LA RIVISTA DI ELETTRONICA APPLICATA, TECNOLOGIE E PRODOTTI

Fare ELETTRONICA

N° 225 - MARZO 2004 - ANNO 20

€ 4,50 - Frs 9,00

TUTORIAL
- PIC® MICROCONTROLLER BY EXAMPLE
- VITAMINA C

HARDWARE
- ALIMENTATORE USB
- SPICE: LA SIMULAZIONE DEI CIRCUITI ELETTRICI ED ELETTRONICI
- CONTATORE MODULARE A DISPLAY
- NOTE SULL'UTILIZZO DI "EASY", IL PROGRAMMATORE DI PICMICRO
- INTRODUZIONE ALLE LOGICHE PROGRAMMABILI

TECNOLOGIE SPERIMENTALI
- SPIRIT, UNA MISSIONE, UN ROBOT
- IL BUS CAN

PRATICAMENTE
- INTERFACCIAMOCI CON LE PORTE LOGICHE (INTERRUTTORE TEMPORIZZATO)

SCUOLABUS
- ITIS "LEONARDO DA VINCI " PISA

MHz
- RICORDO DEI BOLLETTINI TECNICI DELLA GELOSO

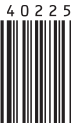
ROBOMANIA
- MANUALE SEMISERIO DI ROBOTICA
- BIOQUADRAPOD: REALIZZAZIONE PRATICA

VITAMINA C

PIC® microcontroller by example

SCUOLABUS

Prima tappa: PISA

SPIRIT, una missione, un robot

Spedizione in A.P. - 45% - ART. 2 COMMA 20/B LEGGE 662/96 - Filiale di Milano. In caso di mancata consegna, restituire all'editore che si impegna a pagare la relativa tassa presso il CMP di Roserio - Milano

ISSN 1591-2272

"MARTE È DIETRO L'ANGOLO..."

Avevo sei anni, quando l'uomo è sbarcato sulla Luna, di quell'evento straordinario ho solo dei vaghi ricordi, ma uno in particolare è rimasto indelebile nella mia mente: mio padre che mi regala un modellino dell'Apollo 11 e mi dice, in parole semplici, che negli anni a venire avrei assistito alla conquista del cosmo, la strada era ormai aperta, era solo una questione di tempo.
Da allora sono passati trentacinque anni e l'uomo, o meglio una sua estensione robotica, sta esplorando Marte alla ricerca di acqua. È incredibile come l'importanza di questo evento sia passato, dopo l'impatto iniziale, quasi in secondo piano; al giorno in cui scrivo, i quotidiani (non tutti per fortuna), danno più evidenza ad un bacio appassionato tra i concorrenti del *Grande Fratello* che alla incredibile conferma che tanto tempo fa su Marte scorreva acqua in cui si sarebbero potute sviluppare primitive forme di vita.
Siamo talmente assuefatti dall'evolversi incessante della tecnologia che un evento come questo diventa quasi "normale". Così mentre il venti Luglio del 1969 milioni di persone erano incollate al video per assistere in diretta a quei primi passi dell'uomo sul nostro satellite naturale, oggi l'esplorazione marziana non ci sconvolge più di tanto.
Ma ci pensate? *Spirit* è su Marte a cinquecento milioni di km di distanza dalla Terra, noi lo controlliamo, gli diciamo cosa fare e come farlo.
Sembra quasi che Marte sia proprio dietro l'angolo...
Ci è sembrato quindi doveroso dedicare a *Spirit*, probabilmente la più alta espressione dell'umana tecnologia, la copertina di questo mese, affidando a *Riccardo Ricci* la stesura dell'articolo.

Come preannunciato nel numero precedente, inizia il viaggio di *Scuolabus* che questo mese fa tappa all'ITIS "Leonardo Da Vinci" di Pisa, il quale presenta un bellissimo progetto sulla gestione di un motore passo-passo. Abbiamo già ricevuto tante richieste di partecipazione, quindi se volete che anche il vostro istituto appaia su queste pagine, scrivete senza esitare a: scuolabus@farelettronica.com.
A giudicare dal numero di messaggi che ci avete inviato, la rubrica *Praticamente* ha centrato il bersaglio, molti di voi hanno risolto il quiz proposto lo scorso mese, questo ci rende davvero felici, è una grande soddisfazione vedere che Fare Elettronica contribuisce all'accrescimento della vostra conoscenza. In questa nuova puntata, *Maurizio Del Corso* spiega l'interfacciamento con le porte logiche e per la parte pratica vi guida alla realizzazione di un interruttore temporizzato.
Abbiamo il piacere di annunciare che, da questo mese, riprende la nostra collaborazione con *Carlo Pria*, segretario dell'AIRE, quindi la conseguente disponibilità, per i lettori appassionati di radio d'epoca, del servizio gratuito per la fornitura di schemi ed informazioni tecniche.
Completano il numero che state per leggere: *Simone Bernardi* con la quarta parte dell'articolo dedicato alle logiche programmabili, *Fabrizio Fazio* con il progetto di un contatore modulare a display (ideale complemento dell'applausometro elettronico presentato precedentemente), *Nico Grilloni* con la terza ed ultima parte del corso su "Spice", ancora *Maurizio Del Corso* con la seconda ed ultima parte dell'alimentatore USB, *Sergio Tanzilli* con la settima parte di "Vitamina C" e la quinta parte di "PIC Microcontroller By Example" (coadiuvato dal sottoscritto), *Nicola Ulivieri* con un trattato sul bus CAN e *Fabio Fioravanzo* che, visto il grande successo di "Easy" il programmatore di PICmicro da lui presentato nel numero 222, risolve i problemi più comuni incontrati dai lettori ed affronta la programmazione in-circuit per la quale Easy è predisposto.
In Robomania trovate la quinta ed ultima parte del "Manuale semiserio di robotica" scritto da *Marco Fabbri* e la seconda parte della realizzazione pratica di "Bioquadrapod" il robot presentato sul numero 223 da *Luigi Carnevale*.
Concludo questo mio intervento mensile introducendo la nuova versione della rubrica dedicata alle fiere e mostre mercato.

Non mi resta che augurarvi una piacevole lettura e vi do appuntamento al prossimo numero in edicola ad Aprile.

Tiziano Galizia
t.galizia@farelettronica.com

FareELETTRONICA www.farelettronica.com

DIRETTORE RESPONSABILE:
GianCarmelo Moroni

DIRETTORE DI REDAZIONE:
Tiziano Galizia (t.galizia@farelettronica.com)

PROGETTO GRAFICO E IMPAGINAZIONE:
Graficonsult - Milano (info@graficonsult.com)

HANNO COLLABORATO:
Nicola Ulivieri, Fabrizio Fazio, Marco Fabbri, Sergio Tanzilli, Maurizio Del Corso, Simone Bernardi, Nico Grilloni, Riccardo Ricci, Fabio Fioravanzo, Carlo Pria, Luigi Carnevale

DIREZIONE - REDAZIONE - PUBBLICITÁ
INWARE srl
Via Cadorna, 27/31 - 20032 Cormano (MI)
Tel. 02.66504794 - 02.66504755 - Fax 02.66508225
info@inware.it - www.inwaredizioni.it

STAMPA:
ROTO 2000
Via Leonardo da Vinci, 18/20 - 20080 Casarile (MI)

DISTRIBUZIONE:
Parrini & C. S.p.a.
Viale Forlanini, 23 - 20134 Milano.
Il periodico Fare Elettronica è in attesa del numero di iscrizione al ROC

UFFICIO ABBONAMENTI
PARRINI & C. S.p.a. Servizio abbonamenti
Viale Forlanini, 23 - 20134 Milano
Per informazioni, sottoscrizione o rinnovo dell'abbonamento:
Telefono: 02.66504794 - Fax: 02.66508225
Email: abbonamenti@farelettronica.com

Spedizione in abbonamento postale 45% art. 2 comma 20/B legge 662/96 - Milano.
Abbonamento per l'Italia: € 39,00
Abbonamento per l'estero: € 99,00
Per la sottoscrizione degli abbonamenti, utilizzare il modulo stampato in ultima pagina.
Gli arretrati potranno essere richiesti, per iscritto, al seguente costo:
Numero singolo: € 7,50
Numero doppio: € 9,00

Autorizzazione alla pubblicazione del Tribunale di Milano n. 647 del 17/11/2003 INWARE srl.

Richieste di assistenza

Per richiedere assistenza o chiarimenti sugli articoli pubblicati, vi preghiamo di contattare l'autore, il cui nome ed indirizzo email è sempre riportato sotto il titolo dell'articolo stesso.
Nel caso ciò non fosse possibile potete scrivere a mailbox@farelettronica.com, ricordandovi di specificare il numero della rivista ed il titolo dell'articolo per il quale chiedete chiarimenti, oltre al vostro nome, cognome ed indirizzo email.
Tutte le richieste con informazioni insufficienti o anonime non saranno prese in considerazione.

Collaborare con Fare Elettronica

La redazione di Fare Elettronica è alla ricerca di collaboratori per la stesura di articoli, progetti, tutorials, rubriche e libri.
Le richieste di collaborazione vanno indirizzate a Tiziano Galizia (t.galizia@farelettronica.com) e accompagnate, se possibile, con una breve descrizione delle vostre competenze tecniche e/o editoriali, oltre che da un elenco degli argomenti e/o progetti che desiderate proporre.

Come contattarci

Indirizzo email della Redazione:
redazione@farelettronica.com
Indirizzo email dell'Ufficio Abbonamenti:
abbonamenti@farelettronica.com
I nostri numeri telefonici:
Telefono 02.66504794 Fax 02.66508225
Il nostro indirizzo postale:
INWARE Edizioni
Via Cadorna, 27/31
20032 Cormano (MI)

Elenco inserzionisti

Artek	27-61
A.R.I. Pescara	45
Blu Nautilus	23-75
Contrive	107
Elettroshop	III cop
Elettroterm	11
ERF	13
Futura	9-33-53
G.P.E. kit	95
Grifo	II cop
Parsic	37-85
Pianeta Elettronica	43
Sandit	110

SOMMARIO

hardware

Introduzione alle logiche programmabili (quarta parte) 14
di Simone Bernardi

Note sull'utilizzo e la programmazione in-circuit con "Easy", programmatore di PICmicro™ 34
di Fabio Fioravanzo

Contatore modulare a display 38
di Fabrizio Fazio

Spice: la simulazione dei circuiti elettrici ed elettronici (terza parte) 46
di Nico Grilloni

Alimentatore USB (seconda parte) 96
di Maurizio Del Corso

praticamente

Interfacciamoci con le porte logiche (interruttore temporizzato) 28
di Maurizio Del Corso

scuolabus

ITIS "Leonardo da Vinci" Pisa 54

tutorial

PIC® Microcontroller By Example (quinta parte) 24
di Tiziano Galizia e Sergio Tanzilli

Vitamina C (Settima parte) 78
di Sergio Tanzilli

tecnologie sperimentali

Il bus CAN 70
di Nicola Ulivieri

Spirit, una missione, un robot (prima parte) 86
di Riccardo Ricci

robomania

Manuale semiserio di robotica (quinta parte): muovere un braccio robotico 100
di Marco Fabbri

Bioquadrapod un robot a quattro gambe: realizzazione pratica (seconda parte) 104
di Luigi Carnevale

Mhz

Ricordo dei bollettini tecnici della Geloso 82
di Carlo Pria

rubriche

Mailbox 6

News 10

Le fiere e mostre mercato di Aprile 2004 76

In Vetrina:

Telecontrollo GSM EL35.002: un "teleassistente" al vostro servizio 111

Dubbi, perplessità, malfunzionamenti, opinioni, commenti o richieste?

Inviateli a: ***mailbox@farelettronica.com***

Oppure scriveta a:
Mailbox - Redazione di Fare Elettronica
Inware srl
Via Cadorna, 27/31 - 20032 Cormano (MI)

Le lettere più interessanti saranno pubblicate in queste pagine.
Per quanto possibile, inoltre, cercheremo di dare una risposta privata a chiunque ci scriverà via email.

D RELÈ DI PICCOLE DIMENSIONI

Salve,come sempre mi congratulo per la novità degli articoli nella rivista. Vi mando questo messaggio per chiedervi chiarimenti su questo problema: ho un circuito con uscita 0/5 V e vorrei collegarci un relé per accendere, quando il segnale è di 5 V, una lampada collegata alla 220 V. Che tipo di relé posso utilizzare in questo caso? Ne esiste uno di dimensioni piccole da circuito stampato?

Francesco Rosati

R Per quanto riguarda i collegamenti elettrici tra l'uscita del suo circuito ed il relé, la rimandiamo alla nuova rubrica "Praticamente…" in cui questo mese si analizzano i circuiti più comuni per interfacciare le uscite delle porte logiche.
Per quanto riguarda il tipo di relé da impiegare, questo dipende dalla corrente assorbita dalla lampada che intende utilizzare. Tale valore si può dedurre dalla potenza della lampada stessa.
Facciamo un esempio: supponiamo di dover accendere una lampada da 220V 25W tramite il relé. Una stima (approssimata per eccesso) della corrente assorbita dalla lampada è data da I=P/V dove P è la potenza e V la tensione di funzionamento della lampada. Nel nostro esempio avremo I=0.12A per cui un relé da 1A sarà più che sufficiente.
Consigliamo di sovradimensionare ampiamente il relé per non incorrere in problemi dovuti al surriscaldamento. Esistono relé da circuito stampato, ma date le loro piccole dimensioni, normalmente non sono adatti a pilotare direttamente carichi alimentati alla tensione di rete. Questi relé possono però pilotare altri relé ai quali verrà applicato il carico.

D COS'È UN TRIMMER

I complimenti sono dovuti, la vostra rivista è veramente interessante. Già che ci sono ne approfitto per farvi una domanda che probabilmente per voi sarà banale, ma che non lo è per qualcuno che non ha fatto corsi per periti elettrotecnici.
Cos'è un trimmer e come funziona? I corsi del Politecnico hanno coperto principalmente come didattica componenti quali diodi, BJTt, MOS e JFET (oltre che ovviamente resistenze, condensatori e induttori). Sarà anche importante sapere come è fatto un 741 ma credo che trascurino poi altri componenti che sono poi quelli che potrebbe capitare di vedere tutti i giorni (se uno si interessa di elettronica ovviamente).
Grazie per l'attenzione dedicatami.

Luca Maccone

R Un trimmer è una resistenza variabile il cui valore può essere impostato spostando il cursore mobile proprio come avviene nei potenziometri. La differenza principale fra un trimmer ed un potenziometro sta soprattutto nelle dimensioni e nella facilità di spostamento del cursore.
Il trimmer è infatti di dimensioni piuttosto ridotte ed il cursore può essere spostato solamente con l'ausilio di un cacciavite. Queste caratteristiche rendono il trimmer un componente assai utile come elemento di taratura, infatti, una volta posizionato il cursore nell'opportuna posizione, risulta praticamente impossibile spostarlo in maniera accidentale. Per sicurezza in alcuni apparati, dopo la taratura, il trimmer viene rico-

perto con un collante che ne impedisce ulteriori manomissioni.
I trimmer possono essere di tipo lineare o logaritmico: nel primo caso la resistenza varia linearmente con lo spostamento del cursore ovvero ruotando il cursore ad esempio di 10° la resistenza varia di una quantità fissa indipendentemente dalla posizione iniziale del cursore. Nei trimmer logaritmici invece, a parità di spostamento del cursore, la variazione di resistenza è maggiore se la posizione iniziale del cursore è vicina ad una delle due estremità.
Esistono inoltre trimmer monogiro e multigiro. Nei primi il cursore può essere spostato per un arco di 120° mentre nei multigiro il cursore si sposta mediante 25 giri di una vite. È evidente che nel secondo caso la variazione di resistenza è più precisa.

D SEGNALE VIDEO

Salve, sto realizzando con i PIC dei sistemi per visualizzare immagini sul televisore attraverso il segnale videocomposito. La mia domanda è questa: alcuni dicono che il segnale deve essere tra 0,3 ed 1 V per l'immagine e di 0 V per i sincronismi, altri dicono che l'immagine deve essere tra 0 e 0,7 V ed invece i sincronismi a -0.3 V. Quale è la verità? Vi ringrazio tanto e buon lavoro.

Giovanni Galati

R Un segnale video nello standard CCIR/PAL, è caratterizzato da 625 linee per ciascuna schermata (frame) e ripete 25 schermate ogni secondo.
Ogni frame è diviso in 2 campi (detti "pari" e "dispari") ciascuno dei quali è composto da 312.5 linee. I campi vengono quindi visualizzati 50 volte al secondo.
Le linee relative ai campi pari-dispari vengono visualizzate in modo alternato (come mostrato nella figura 2) e questa tecnica è identificata con il termine "interlacing" (interlacciamento).
L'interlacciamento viene usato per ridurre l'effetto di lieve intermittenza nella luminosità dell'immagine visualizzata. Il processo di esame intrecciato è indicato in figura 2.

L'inizio di una linea orizzontale è contrassegnato da un impulso di sincronizzazione orizzontale mentre l'inizio di ogni campo è contrassegnato da un impulso di sincronizzazione verticale.
Per quanto visto finora in ogni frame ci sono 625 im-

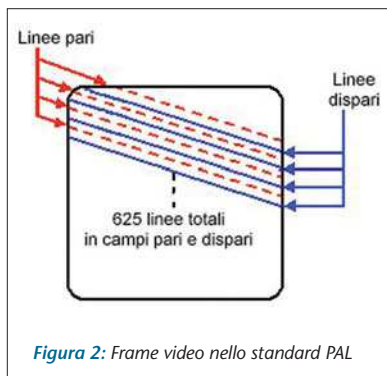



Figura 2: Frame video nello standard PAL

pulsi di sincronizzazione orizzontale e 50 impulsi di sincronizzazione verticale per ogni secondo. Gli impulsi di sincronizzazione assicurano che l'immagine sia perfettamente bloccata sullo schermo.
Per segnali video monocromatici, l'immagine è costituita da informazioni sulla luminanza (Y), 0 V è il livello del nero, mentre 0,7 V corrisponde al bianco. Gli impulsi di sincronizzazione orizzontale hanno ampiezza di 0,3 V negativi. Il segnale video relativo ad una linea ha quindi la struttura di figura 3.

Figura 3: Struttura del segnale video relativo ad una linea

D BATTERIE RICARICABILI DA 1,2 V

Devo alimentare un circuito auto-costruito con delle batterie ricaricabili, ma non riesco a trovarne da 1,5 V. In pratica ho bisogno di raggiungere la tensione di 4,5 V, ma utilizzando 3 batterie al Ni-Cd da 1,2 V raggiungo soltanto 3,6 V, invece, utilizzandone 4 raggiungo 4,8 V. Insomma non riesco a centrare la tensione che mi serve. Perché non si trovano batterie ricaricabili da 1,5 V ma solo da 1,2 V?
Continuate così Fare elettronica è una rivista molto interessante.

Francesca Amodio

R Il motivo sta nella reazione chimica che genera la differenza di potenziale: nelle pile alcaline l'elemento base genera 1,5 V mentre nelle ricaricabili Ni-Cd è 1,2 V.
Tutti i valori sono multipli dei valori base. Non è detto quindi che se un dispositivo funziona con batterie alcaline funzioni anche con batterie ricaricabili. Tutto dipende dalla tolleranza che il dispositivo ammette sulla tensione di alimentazione. Provare non è comunque distruttivo se la differenza si mantiene intorno a 0,5 V: al più il dispositivo non si accenderà.
Le batterie ricaricabili sono comunque migliori per quanto riguarda la capacità di erogare corrente infatti si possono superare anche i 2 Ah (ovvero la

batteria è in grado di erogare 2 A di corrente per un ora) contro i decimi di Ampere delle normali pile alcaline.

SENSORE DI TEMPERATURA

Spettabile Fare Elettronica, sfogliando la rivista 202 (Aprile 2002), mi sono soffermato ad analizzare un po' nel dettaglio il funzionamento del modulo termometro (-0°/150°C) a pagina 82. Per curiosità ho anche scaricato il datasheet del sensore LM35CZ (dal sito della National Semiconductor), ma mi sono accorto, controllando la piedinatura, che il collegamento del sensore nella foto della rivista (paina 84 figura 4), è errato. Infatti il terminale + è stato collegato a massa e viceversa. Sto sbagliando io oppure è cosi?

Pier Magliani

Il sensore di temperatura LM35CZ è un dispositivo che fornisce una tensione in uscita, proporzionale alla temperatura dell'ambiente in cui si trova immerso.
In particolare, la tensione di uscita si incrementa di 10mV per un aumento della temperatura di 1°C. La pedinatura del sensore è quella di figura 4, per cui le connessioni riportate nel progetto indicato dal lettore, sono corrette. Si faccia attenzione alla scritta "bottom view" che indica la vista dal basso del dispositivo.

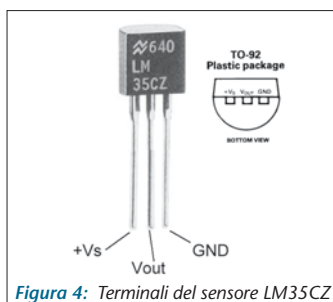



Figura 4: Terminali del sensore LM35CZ

SENSORE DI PROSSIMITÀ

Per arricchire un piccolo robot di mia costruzione, vorrei realizzare un sensore di prossimità impiegando un led ad infrarossi ed un fototransistor, in modo da eccitare un relé nel caso in cui il robot rilevi un ostacolo. Avete uno schema da suggerirmi? Grazie per la vostra risposta.

Lorenzo Ferrari

È possibile realizzare un sensore di prossimità, utilizzando il circuito in figura 5.
Quando il fototransistor è illuminato la tensione sul morsetto invertente si abbassa e, se scende sotto il livello impostato per quella sul morsetto non invertente, l'uscita dell'operazionale va a livello alto ed il relé si eccita.
Il circuito funziona ad alimentazione singola.
Se vuole agire sulla sensibilità del fototransistor può utilizzare lo schema in figura 6.

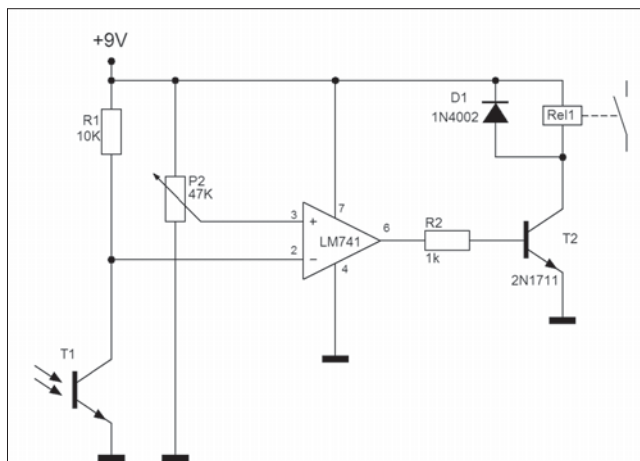



Figura 5: Schema del sensore di prossimità

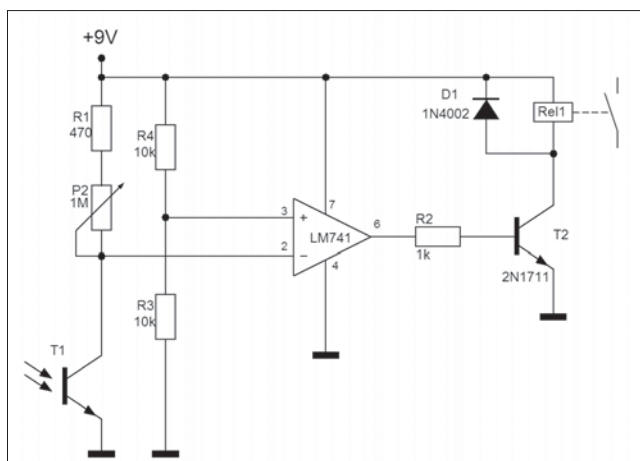



Figura 6: : Schema del sensore di prossimità con regolazione della sensibilità

PANRADIO MOD.111

Mi rivolgo a voi per avere notizie su di un ricevitore che ho nella mia collezione. Si tratta di un apparecchio il cui mobile ha lo stile tipico della produzione a cavallo fra la fine degli anni '40 ed i primi anni '50. Su di una targhetta è riportata l'indicazione "Panradio mod.111".
Chiedo a voi notizie in merito a questa marca oltre che lo schema elettrico del ricevitore per procedere al restauro.

Renato Bernardi

Il Panradio mod.111 è stato costruito negli anni 1946/47 dalla FIMI/Phonola per conto della Refit.
La Refit era una organizzazione commerciale, con sede a Roma, che distribuiva apparecchi radio con i marchi Kosmovox e Panradio, la cui costruzione era affidata alla FIMI/Phonola di Saronno. Si tratta di apparecchi costruiti con la cura e la qualità tipiche di questo fabbricante. La produzione di questi marchi risulta cessata intorno alla metà degli anni '50.
Per posta Le abbiamo inviato lo schema elettrico dell'apparecchio.

Registratori digitali multimediali audio/video

I Video Recorder Archos raggruppano in un'unica soluzione un lettore/registratore video MPEG-4 modulare, un lettore/registratore Mp3, un portafoto digitale e un dispositivo di memorizzazione dati, il tutto in un oggetto che sta nel palmo della mano!

REGISTRATORE PORTATILE A/V su HARD DISK

Euro **495,00**

AV140

Il Video Recorder AV140 è il primo personal entertainment center disponibile sul mercato: utilizzabile sia come sistema multimediale che in applicazioni destinate al settore della sicurezza. Si compone di una unità di memorizzazione/visualizzazione e di un convertitore video esterno con telecomando (JBM Digital Video Recorder) compreso nella confezione. Il sistema registra filmati in MP4 in maniera semplicissima da qualsiasi sorgente: TV, videoregistratori e lettori, telecamere. Mentre un normale nastro registra al massimo 2-4 ore, l'AV140 ha una capacità di registrazione fino ad 80 ore (in funzione della risoluzione e della qualità di registrazione). I files MP3 vengono registrati da qualsiasi sorgente audio; inoltre, tramite il microfono incorporato è possibile registrare la voce con una capacità di registrazione di oltre 2.000 ore! L' AV140 può memorizzare fino a 400,000 foto, con la possibilità di visualizzarle sull'LCD incorporato o direttamente sulla TV. L'hard disk interno da 40 Gb può essere utilizzato come hard disk aggiuntivo per il PC: collegato tramite l'USB 2.0 alla porta USB del PC viene "visto" come un hard disk supplementare.

CARATTERISTICHE TECNICHE

Capacità dati: 40 Gb; **LCD:** Incorporato TFT risoluzione 237x234; **Interfaccia:** USB 2.0 (PC & Mac) (compatibile 1.1) cavo firewire opzionale; **File supportati:** MP3 (30 to 320 Kbps CBR and VBR) reading, MP3 VBR (from 30 to 160 kbps) real-time encoding. Photo Carousel JPEG e BMP), compatibile MPEG4 video con DivX (CIF 352x288 pixels, 25 frames/sec., MP3 sound track VHS Quality). Autonomia fino a 7 ore; **Funzione Musicale:** Lettore Musicale Stereo MP3 decoding @ 30-320 Kbits/s. CBR & VBR, Registrazione digitale Stereo MP3 encoding @ 30-160 Kbits/s. VBR - Compressione audio in registrazione (con DVR) Stereo MP3 96-192 kb/s CBR (Constant Bit Rate); **Playback Video:** MPEG-4 SP con suono MP3 stereo qualità DVD. Risoluzione fino a 640x368 @25f/s. formato AVI, legge XviD e DivX 4.0 & 5.0 - Modulo DVR per la registrazione con compressione video MPEG4-SP; **Funzione Visualizzazione Foto:** JPEG e BMP qualsiasi dimensione; **Connessioni AV:** Linea Stereo analogica in Ingresso e Digitale (Ingresso e Uscita), Video composito in/out,S-VIDEO (su modulo DVR), cuffie, microfono incorporato; **Compatibilità PC:** Pentium II 266 MHz o superiori. Windows 98 SE, ME, 2000, XP, 64 MB RAM. Mac: 9.2 o 10.2.4 iMac, G3 o superiore; **Accessori inclusi:** AV140 Video Recorder, cavo USB 2.0, Alimentatore, batterie al litio, cuffie stereo, cavo AV (SCART). CD con MusicMatch, drivers e manuale in 6 lingue (italiano incluso). Digital Video Recorder, cavi audio & video, telecomando, registratore DVR; **Software:** CD con MusicMatch, software vari Archos; **Dimensioni:** 110 x 79 x 30 mm (modulo DVR 60 x 54 x 30mm 45 grammi); **Peso:** 290 gr.

AV340 - Registratore portatile su HARD DISK

Si differenzia dal modello precedente per l'impiego di un monitor da 3,8" che consente una migliore visione di immagini e filmati. Anche in questo caso è presente un convertitore video esterno con telecomando (JBM Digital Video Recorder) per l'acquisizione dei filmati. Il sistema registra filmati in MP4 in maniera semplicissima da qualsiasi sorgente: TV, videoregistratori e lettori, telecamere con una capacità di registrazione fino ad 80 ore (in funzione della risoluzione e dalla qualità di registrazione). I files MP3 vengono registrati da qualsiasi sorgente audio con una capacità di registrazione di oltre 2.000 ore! L'AV340 può memorizzare fino a 400,000 foto, con la possibilità di visualizzarle sul monitor incorporato o direttamente sulla TV. L'hard disk interno da 40 Gb può essere utilizzato come hard disk aggiuntivo per il PC: collegato tramite l'USB 2.0 alla porta USB del PC viene "visto" come un hard disk supplementare.

CARATTERISTICHE TECNICHE

Capacità dati: 40 Gb; **Monitor:** 3,8" (QVGA), 320x240 pixels; **Interfaccia:** USB 2.0 (PC & Mac, compatibile 1.1) cavo firewire opzionale; **File supportati:** MP3 (30 to 320 Kbps CBR and VBR) decoding MP3 VBR (from 30 to 160 kbps) encoding; **Visualizzazione di foto e video:** BMP & JPEG (qualunque dimensione), MPEG-4 SP; **Funzione Musicale:** Lettore Musicale Stereo MP3 decoding @ 30-320 Kbits/s. CBR & VBR; Playback Video: MPEG-4 SP con audio MP3 stereo, simile alla qualità DVD. **Risoluzione:** 352x288@30f/s, o 640x272@25f/s, fino a 640x368@20f/s; Formato file: AVI, legge i formati XviD e DivX 4.0 & 5.0. *Dotato di modulo DVR per la registrazione da qualsiasi fonte video.* **Caratteristiche modulo DVR 100:** Video Input: video composito o S-Video, **Audio Input:** audio stereo mini-jack o RCA; **Compressione Video:** Mpeg-4 SP con audio stereo MP3; **Compressione Audio:** stereo MP3 96-192 kb/s CBR; Dimensioni e peso: 60 x 54 x 30mm, 45g; **Connessioni AV:** Linea Stereo analogica in ingresso e Digitale in ingresso ed uscita. Video composito in/out, S-VIDEO (su modulo DVR), cuffie, microfono incorporato; **Compatibilità PC:** Pentium II 266 MHz o superiori. Windows 98 SE, ME, 2000, XP . 64 MB RAM . Mac: 9.2 o 10.2.4 iMac, G3 o superiore; **Accessori inclusi:** AV340 Video Recorder, cavo USB 2.0, alimentatore, batterie al litio (già installate), cuffie stereo, cavo AV (SCART). CD con MusicMatch, drivers e manuale in 6 lingue (italiano incluso); Digital Video Recorder, cavi audio & video, telecomando, registratore DVR; Software: software vari Archos; **Informazioni aggiuntive:** Firmware aggiornabile tramite download da web, autonomia: fino a 10 ore se utilizzato per l'ascolto di file audio (MP3) e circa 3 ore e mezza se utilizzato per la visualizzazione di video sul display LCD incorporato; **Dimensioni:** 112 x 82 x 31 mm; **Peso:** 350 gr.

AV380 - Registratore portatile su HARD DISK (80 Gb)

Stesse caratteristiche del modello AV340 ma con hard disk da 80 Gb e capacità di memoria doppia.

AV380

Euro **995,00**

Prezzi IVA compresa.

V. le Kennedy, 96 - 20027 Rescaldina (MI) - Tel. 0331/576139
Fax. 0331/466686 - *www.futuranet.it*

Caratteristiche tecniche e vendita on-line:
www.futuranet.it

NEWS

412

*Questo spazio è gentilmente offerto da **EONews**, il Quindicinale di notizie e commenti per l'industria elettronica di **VNU Business Publications Italia**.*

Quindicinale di notizie e commenti per l'industria elettronica

EO ELETTRONICA OGGI NEWS

all'interno

Debole richiesta di semiconduttori

All'interno della supply chain solo i distributori sorridono

Avanti, con calma

IDEE DI PROGETTO: LA NUOVA INIZIATIVA SUL WEB DI ELETTRONICA OGGI

"Idee di progetto – Design Ideas" è la nuova rubrica di Elettronica Oggi che è stata lanciata dal mese di Settembre 2003. Caratteristica saliente di questa nuova iniziativa è che sarà completamente ed esclusivamente on line ed accessibile dal sito www.ilb2b.it.
L'obbiettivo principale è creare una vera e propria libreria di idee alla quale tutti coloro che operano in maniera professionale nel mondo dell'elettronica possano "catturare" informazioni e suggerimenti utili per il loro lavoro quotidiano. Questo nuovo strumento vi permette di scaricare, con un solo click, tutte le risorse necessarie per risolvere velocemente qualsiasi problema e, in ultima analisi, minimizzare il time to market.

SENSORE DI TEMPERATURA

Cesvit Microelettronica ha sviluppato un sensore che sfrutta le potenzialità combinate della sensoristica e delle tecniche di identificazione in radiofrequenza. In particolare, nel progetto denominato MAMA (Monitoring fish for manufacturing process with Advanced Miniaturised subsystem and radiofrequency Apparatus) l'azienda ha messo a punto un sensore di temperatura integrato in un transponder a 13,56 MHz. Il sistema, attualmente in uso per l'azienda Circeo Pesca, consente di ottenere un reporting completo e affidabile sulla storia termica del prodotto, accesso in tempo reale ai dati di produzione e monitoraggio continuo delle temperature operative. Il dispositivo è in grado di memorizzare valori di temperatura nel range di –20, +70 gradi C a intervalli di tempo programmabili, con un'autonomia massima di 3 settimane di operatività.

readerservice.it ***412 - 51***

MICROCONTROLLORE CON MEMORIA FLASH EMBEDDED

È disponibile da Claitron H8S/2377/F, il nuovo microcontrollore a 16 bit di Renesas con memoria Flash embedded, progettato per ridurre costi, tempi di sviluppo, ingombri e consumi dei sistemi complessi, aumentandone al tempo stesso l'affidabilità. È stato sviluppato per il mercato Epos, ma si presta all'impiego in molte altre applicazioni consumer e industriali.
Opera a 3,3 V alla frequenza di 33 MHz con circa 15 Mips Dhrystone. Il suo assorbimento tipico è di soli 80 mA. È dotato di 384 kbyte di memoria Flash e 24 kbyte di Ram. Esiste anche una versione Rom mascherata a basso costo, particolarmente indicata per realizzare la piattaforma Stip con Jeff.

readerservice.it ***412 - 52***

FILTRO EMI ATTIVO PER IL MERCATO DEI BUS TELECOM

Vicor ha presentato il nuovo filtro EMI attivo QPI-1 per applicazioni con convertitori DC-DC da 48V. Il QPI-1 mette a disposizione oltre 40dB di attenuazione del rumore di modo comune e più di 80 in modalità differenziale a 500kHz, con un montaggio in superficie da 24,5 x 24,5 x 5,1mm. Il QPI-1 è in assoluto il primo filtro EMI attivo destinato al mercato dei bus Telecom, e offre dal 50 all'80% di risparmio di spazio su scheda rispetto ad altre soluzioni. Il filtraggio attivo elimina il ringing in ingresso al convertitore DC-DC e migliora la risposta ai transienti di carico e di linea. A differenza delle soluzioni passive, il QPI-1 attenua il rumore sull'intera gamma di frequenze. Inoltre, non vi sono elementi risonanti che possano amplificare il rumore stesso. Il QPI-1 risponde alle specifiche del bus Telecom internazionale da 36 a 75V CC, incluso il picco da 100 V – 100 ms.

readerservice.it ***412 - 56***

DRIVER MOSFET

Ltc4440 di Linear Technology (Silverstar Celdis) è un driver Mosfet a canale N high side tollerante ai transitori con gate driver che minimizzano perdite di commutazione in alimentatori. Alloggiato in package ThinSot e Msop a 8 lead, il funzionamento a 80 V (transitori 100 V), corrente di uscita pull-up di picco di 2,4 A e impedenza di uscita pull-down di 1,50 ohm lo rendono perfettamente adatto per pilotare Mosfet di soglia standard a corrente, tensione e capacità di gate elevate. Migliora il rendimento e la capacità di potenza in uscita di convertitori Dc/Dc riducendo, rispettivamente a 7 e 10 ns, i tempi di salita e discesa di gate Mosfet allorché pilotino carichi di 1.000 pF. Perfeziona le prestazioni di alimentatori in sistemi telecom 48 V

come anche in sistemi a potenza distribuita utilizzati in applicazioni automotive e industriali.

readerservice.it 412 - 60

FUSIBILE MONOUSO

Raychem Circuit Protection, divisione di Tyco Electronics, ha annunciato il suo primo fusibile monouso per applicazioni telecom e networking. La nuova serie di fusibili Ft600 è progettata per aiutare i produttori di apparecchiature per le telecomunicazioni nel rendersi conformi agli standard statunitensi in materia di sicurezza per le situazioni di sovraccorrente, inclusi Telcordia Gr-1089, Fcc Parte 68 e la terza edizione di Ul60950.

readerservice.it 412 - 63

FAMIGLIA DI PLL PER APPLICAZIONI AD ALTA FREQUENZA

Fujitsu Microelectronics ha annunciato la serie MB15F7xUV, una nuova famiglia di sintetizzatori di frequenza PLL doppi sub miniaturizzati progettata per il mercato delle comunicazioni mobili ad alta frequenza. Rispondendo alla domanda di miniaturizzazione espressa dal mercato, il progetto di questa nuova generazione di PLL Fujitsu ha permesso di alloggiare i dispositivi nel contenitore per PLL più piccolo attualmente disponibile. Si tratta dell'innovativo package Bump Chip Carrier (BCC) a 18-pad, caratterizzato da un ingombro di soli 2,4x2,7x0,45 mm.
Questo package consente di montare i dispositivi in un'area pari a quasi la metà di quella occupata dai PLL più piccoli attualmente in commercio, a fronte di un'altezza inferiore del 40% rispetto alle soluzioni convenzionali.

readerservice.it 412 - 66

CONVERTITORI ANALOGICO-DIGITALI QUADRUPLI

I convertitori analogico-digitali quadrupli Ad9229 e Ad9289 di Analog Devices sono integrati in un chip allo scopo di soddisfare le necessità dei sistemi vincolati da problemi di spazio, ad alta densità, quali sistemi di imaging medicale, o applicazioni multicanale, quali stazioni base di comunicazioni wireless. Utilizzando uscite di dati seriali a segnalazione dif-

ferenziale, a bassa tensione, si è riusciti a incorporare quattro Adc su un singolo chip, creando una soluzione compatta, veloce, competitiva in termini di costi e efficiente in fatto di potenza, riducendo drasticamente conteggio dei pin, dimensioni dei package, numero di tracce di scheda e rumore del substrato.

readerservice.it 412 - 69

CONTROLLORI DC/DC

Micrel (Aertronica) ha progettato e sviluppato una famiglia di cinque nuovi controllori Dc/Dc (Mic2198/9/3/4/6) ad alta efficienza e semplici da utilizzare, alloggiati in piccoli package a basso numero di pin.
Mic2198/9 sono ideali per alimentare l'ultima generazione di chipset impiegati nelle odierne architetture Dc/Dc a punto di carico.
Funzionano con tensioni di ingresso da 4,5 a 32 V, rendendoli adatti per usi con sistemi a bus di potenza a 5, 12 e 24 V.
Mic2193 è un buck controller sincrono in grado di conseguire duty cycle fino al 100%, mentre Mic2194 è un buck controller flessibile ottimizzato per applicazioni buck "step-down".
Mic2196, infine, è un boost "step-up" controller capace di generare tensioni e correnti a uscita elevata.

readerservice.it 412 - 70

DISPOSITIVI FPGA NON VOLATILI

Actel ha annunciato la disponibilità di dispositivi Fpga non volatili ad alte prestazioni, qualificati per specifiche industriali. Il funzionamento dei nuovi dispositivi Axcelerator a chip singolo di grado industriale, con densità fino a 2 milioni di gate, É ora garantito con temperature ambiente comprese tra -40 e +85 gradi centigradi. Con queste nuove soluzioni, Actel può fornire all'industria una completa linea di soluzioni a basso consumo. L'estesa gamma di temperature, unite a ottime prestazioni, sicurezza di progetto e immunità dal firm-error, li rende adatti per applicazioni alternative agli Asic per comunicazioni di dati, telecomunicazioni, networking, collaudo, misura e controllo industriale.

readerservice.it 412 - 74

CONVERTITORE ANALOGICO-DIGITALE

Atmel ha annunciato la disponibilità in volumi di produzione di una nuova versione del proprio convertitore analogico-digitale finalizzato a ricevitori satellitari in banda larga, radar/Ecm e applicazioni di strumentazione di fascia alta. Siglato Ts83102G0B, combina risoluzione 10 bit, velocità di campionamento 2 Gsp max, larghezza di banda 3,3 GHz in un compatto package Cbga. Fornisce ottime prestazioni consumando solo 4,6 W. L'Sdfr è di 60 dBfs a frequenze di ingresso di 1,4 Gsps/700 MHz e sempre nel campo di 55dBfs a frequenze di ingresso di 2 Gsps/2 GHz. La distorsione di intermodulazione del terzo ordine a due toni è di 65 dB a 1,4 Gsps su una banda 500 MHz centrata attorno 1 GHz, consentendo di digitare segnali in elevata banda larga If con canali adiacenti con minimi componenti di spettro parasitici.

readerservice.it 412 - 75

SMART CARD A 32 KB

Samsung Electronics ha presentato nel corso di Cartes 2003 a Parigi la sua nuova smart card a 32Kb che combina insieme interfacce contact e contactless. La doppia interfaccia permette di accedere a una più ampia gamma di servizi e, consente a chi li eroga di trarre benefici da una più ampia base di lettori di card. Con una EPROM di 32Kb, la card è capace di supportare anche un "public key cryptography engine" che ne aumenta la sicurezza. Con questo dispositivo Samsung ha fornito la risposta alle richieste di applicazioni sicure, con un sistema operativo capace di supportare molteplici applicazioni. Infatti, la configurazione permette di supportare applicazioni come Java Virtual Machine e le relative applets, consentendo alla nuova tecnologia di crescere anche nei mercati con contenuti a valore aggiunto.

readerservice.it 412 - 78

COME OTTENERE MAGGIORI INFORMAZIONI

EONews offre il servizio "reader service" che vi consente, utilizzando l'apposito codice riportato alla fine di ogni news, di ricevere maggiori informazioni.

Visitate il sito **www.readerservice.it** e compilate la cartolina virtuale con i vostri dati, il numero della rivista, questo mese il **412**, ed i numeri di reader service presi delle notizie che vi interessa approfondire.

EONEWS provvederà, tempestivamente, a contattare le aziende interessate, che invieranno al vostro indirizzo tutta la documentazione disponibile.

MARCHEFIERE
ERF ENTE REGIONALE PER LE MANIFESTAZIONI FIERISTICHE

INTRODUZIONE ALLE LOGICHE PROGRAMMABILI

quarta parte

di Simone Bernardi
artt@iol.it

Ovvero: gran finale. Una scheda di sviluppo per CPLD Altera, su cui sperimentare a piacimento ogni possibile circuito digitale. Contatori, decodifiche, look-up table, timer, PWM controller, una lista infinita di applicazioni senza doversi preoccupare di nient'altro che di amministrare la propria fantasia e creatività.
Dopo i voli tranquilli delle puntate precedenti e le prime manovre di alta quota sul Tool di progettazione e simulazione MAX II Plus, è il momento di eseguire qualche vera acrobazia, e riscuotere gli applausi...

MINIMAX-44

Negli ultimi tempi ho seguito lo sviluppo di alcune schede che utilizzavano CPLD Altera; le prime volte mi sono messo l'anima in pace e ho cablato i prototipi su una basetta millefori, poi ho deciso che ne avevo abbastanza ed ho acquistato la scheda di sviluppo descritta in queste pagine. La MiniMAX-44, visibile in figura 1, consente di avere a disposizione tutti gli ingressi e le uscite di una CPLD a 44 pin, senza doversi preoccupare del cablaggio e dei problemi di alimentazione. La scheda comprende quindi essenzialmente un alimentatore che permette di selezionare 5 V o 3.3 V, i condensatori di bypass sui pin di alimentazione del chip, una serie di connettori a pettine attraverso i quali si può accedere ai pin di I/O della CPLD; questi ultimi, infine, sono provvisti di opportuni pull-down resistivi.

Figura 1: *La scheda MiniMAX-44*

La MiniMAX-44 è intenzionalmente priva di orpelli, quali interfacce, display, memorie, che sono spesso montati sulle schede di sviluppo per logiche programmabili; gli unici "dispositivi" di I/O previsti sono 4 pulsanti, 4 dip-switch e 8 led, tutti escludibili. Come conseguenza la scheda è semplice, affidabile, flessibile e, soprattutto, economica.

La MiniMAX-44 può ospitare le CPLD Altera:
- EPM3032ALC44
- EPM3064ALC44
- EPM7032SLC44
- EPM7064SLC44

di cui ci siamo occupati in dettaglio nella seconda puntata, ed è stata realizzata tenendo conto delle due principali differenze tra i dispositivi MAX70XX e i MAX30XX; i primi, infatti, funzionano a 5 V mentre i secondi a 3.3 V ed

Figura 2: Schema elettrico

è quindi stato previsto un jumper (JP1) per selezionare le due diverse alimentazioni. Le MAX70XX e le MAX30XX sono pin-to-pin compatibili tranne che per i pin 17 e 36, che sono generici pin di I/O per le prime e pin di alimentazione (gnd) per le seconde; dunque sono stati previsti due jumper (JP2 e JP3) che permettono di assegnare questi due pin in maniera opportuna.

Lo schema elettrico

In figura 2 è riportato lo schema elettrico della scheda MiniMAX-44. Sulla destra è visibile la sezione dell'alimentatore, che fa capo ad un LM317 la cui rete di regolazione può essere determinata in funzione del jumper JP1: se il jumper viene lasciato **aperto**, la tensione VCC è pari a 5V, se il jumper viene **inserito**, tale tensione è invece 3.3V.

Sotto il gruppo di alimentazione si trova il connettore a 10 poli della porta JTAG di programmazione, con le resistenze di pull-up R6 e R7 sulle linee TMS e TDI, e la resistenza di pull-down R8 sulla linea TCK; ci siamo occupati in dettaglio di queste linee nella seconda puntata.

In alto a sinistra c'è il primo connettore a strip CON1 con i pin di I/O

CON1	CON2	CON3	CON4	CON5
gnd	gnd	gnd	gnd	gnd
vcc	vcc	vcc	vcc	vcc
43-I/glclk	**14**-I/O (S1)	**41**-I/O (LED8)	**39**-I/O	**26**-I/O
1-I/glclr	**12**-I/O (S2)	**40**-I/O (LED7)	**37**-I/O	**25**-I/O
44-I/oe1	**11**-I/O (S3)	**24**-I/O (LED6)	**34**-I/O	**17***-I/O/gnd
2-I/oe2	**9**-I/O (S4)	**21**-I/O (LED5)	**33**-I/O	**36***-I/O/gnd
	8-I/O (S5/a)	**20**-I/O (LED4)	**31**-I/O	
	6-I/O (S5/b)	**19**-I/O (LED3)	**39**-I/O	
	5-I/O (S5/c)	**18**-I/O (LED2)	**28**-I/O	
	4-I/O (S5/d)	**16**-I/O (LED1)	**27**-I/O	

Tabella 1: Numerazione del pin I/O sui connettori

glclk, glclr, oe1 e oe2. Nella parte inferiore dello schema sono visibili gli altri quattro strip CON2÷CON5. La lista dei pin disponibili per ognuno di essi è riportata nella tabella 1. Tutti i pin di I/O sono provvisti di resistenze di pull-down da 47 KΩ.

Tutti i pin possono essere configurati individualmente come ingresso o uscita, tranne quelli disponibili sul CON1, ovvero 1, 2, 43 e 44, che possono essere usati solo come ingressi. Come già detto i pin 17 e 36 (*) sono pin di I/O per le MAX70XX mentre sono pin di massa per le MAX30XX e quindi, in quest'ultimo caso, devono essere inseriti i jumper JP2 e JP3 proprio per collegarli a massa.
ATTENZIONE: utilizzando le MAX70XX i jumper JP2 e JP3 devono essere **scollegati**.

Ai pin facenti capo al CON2 è possibile collegare i pulsanti S1÷S4 e il quadruplo dip-switch S5, **inserendo** il jumper JP4; **scollegando** tale jumper, al contrario, i pulsanti non risultano alimentati e quindi i pin di I/O sono di fatto scollegati dagli switch stessi. In quest'ultimo caso, si consiglia di tenere comunque aperti i quattro dip-switch di S5. La rete RN6 è stata prevista per proteggere la CPLD nel caso venisse premuto un pulsante connesso ad un pin configurato come output. Ai pin facenti capo al CON3, invece, è possibile collegare i led DL1÷DL8, **inserendo** il jumper JP5; **scollegando** tale jumper, i led non risultano alimentati e quindi i pin di I/O sono di fatto scollegati dai led stessi.
ATTENZIONE: ogni led risulta connesso tra Vcc e il pin della CPLD quindi si accende se il pin di uscita corrispondente è a livello logico "0" e si spegne se è a livello logico "1", Per finire, alla sinistra del chip sono visibili i 4 condensatori di bypass, C1, C2 C3 e C4, che nella scheda sono montati accanto ai pin di alimentazione della CPLD.

Le differenti impostazioni dei jumper in base alla CPLD montata sulla scheda sono riassunti in tabella 3.

Sigla	Valore
PWR_PLUG	Jack alimentazione da cs
C1÷C4, C6, C8	0,1 µF poliestere
C5	220 µF 25 V elettrolitico
C7	10 µF 25 V elettrolitico
CON1, CON5	Strip Femmina 6 poli
CON2÷CON4	Strip Femmina 10 poli
D1	1N4004
IC1	LM317T
JP1÷JP5	Jumper
JTAG	Connettore Panduit Maschio da cs 10 poli
LD1÷LD8	Led 3mm rosso
LD9	Led 3mm verde
PLCC44	Zoccolo PLCC44
R1, R3, R6÷R8	1 KΩ 1/4 W
R2	330 Ω 1/4 W
R4	1,2 KΩ 1/4 W
R5	22 KΩ 1/4 W
R5	22 KΩ 1/4 W
RN1÷ RN5	47 KΩ Rete resistiva 8+1
RN6	1.5 KΩ Rete resistiva 8+1
RN8	47 KΩ Rete resistiva 4+1
S1÷S5	Pulsante omron n.a. da cs
S5	Dip-switch quadruplo

Tabella 2: *Lista componenti della scheda MiniMAX-44*

	JP1	JP2	JP3
MAX30XX (EPM3032ALC44, EPM3064ALC44)	CHIUSO	CHIUSO	CHIUSO
MAX70XX (EPM7032SLC44, EPM7064SLC44)	APERTO	APERTO	APERTO

Tabella 3: *Impostazione dei Jumper sulla scheda*

Per alimentare la MiniMAX-44

Per alimentare la scheda è sufficiente utilizzare un qualunque alimentatore a spina che sia in grado di erogare una tensione continua di 9-12 V; l'assorbimento della scheda dipende dalla frequenza di lavoro e dalle schede aggiuntive che posso-

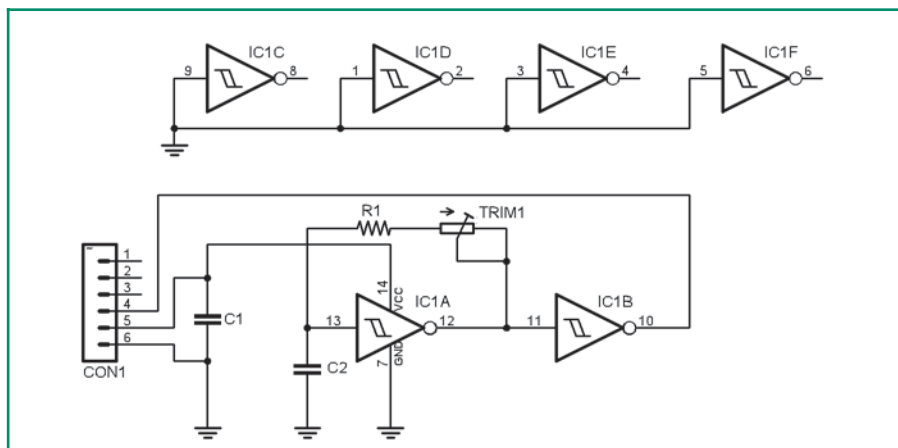



Figura 3a: *Schema elettrico scheda di clock*

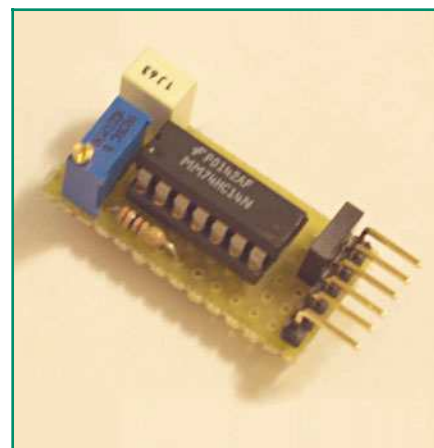



Figura 3b: *Scheda di clock*

no essere collegate sui connettori a strip; per evitare il surriscaldamento dell'integrato LM317, è bene non superare i 2-300 mA di assorbimento e quindi fate riferimento a tale valore per la scelta dell'alimentatore. Il jack deve avere il polo positivo sul conduttore centrale.

Il segnale di clock

Al contrario di un microprocessore, una logica programmabile non ha necessariamente bisogno di un segnale di clock per funzionare. Se ad esempio si vuole implementare una decodifica, gli unici segnali necessari sono quelli del bus di ingresso e quelli del bus di uscita. Ci sono tuttavia numerose applicazioni, come quelle descritte nelle pagine seguenti, per le quali un segnale di clock è indispensabile; la MiniMAX-44 non possiede un oscillatore a bordo, quindi questo segnale deve essere fornito dall'esterno, usando, per esempio, un generatore di onde quadre, oppure un piccolo plug-in come quello in figura 3/a e 3/b.

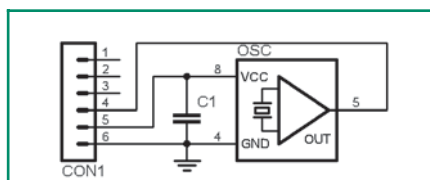



Figura 4a: *Schema elettrico scheda di clock integrata*

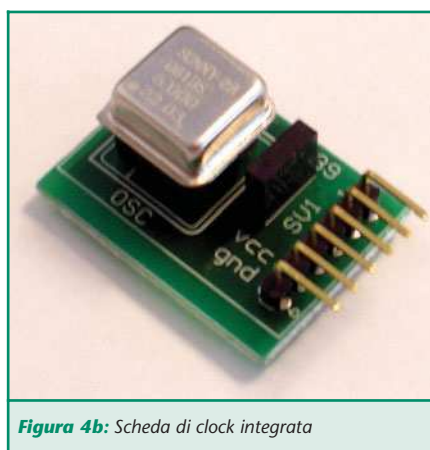



Figura 4b: *Scheda di clock integrata*

Si tratta di un circuito molto semplice, che utilizza porte NOT a trigger di Schmitt in una classica configurazione di retroazione; la lista dei componenti è riportata nella tabella 4. Con i valori indicati, attraverso il TRIM1, si può variare la frequenza da un minimo di 10 Hz a un massimo di 8 KHz circa; basta sostituire il condensatore C2, comunque, per ottenere intervalli diversi.

Sigla	Valore
C1	0,1 µF poliestere
OSC	Oscillatore DIP8 1÷50 MHz
CON1	Connettore strip 90° M 6 pin

Tabella 5: *Lista componenti scheda di clock integrata*

Sigla	Valore
R1	100 Ω 1/4 W
TRIM1	100 KΩ
C1	0.1 µF poliestere
C2	1 µF poliestere
IC1	74HC14
CON1	Connettore strip 90° M 6 pin

Tabella 4: *Lista componenti scheda di clock*

Attenzione alla scelta dell'integrato, che deve essere necessariamente del tipo HC per poter funzionare anche a 3.3 V.

La schedina è stata pensata per poter essere inserita indifferentemente nella connettore CON1 oppure nel connettore CON5; come già detto, applicare il segnale di clock nella strip CON1, e quindi alla linea di Global Clock (pin 43), ha senso solo se si implementano progetti sincroni. Poiché questa applicazione è riservata a mani esperte, nella maggior parte dei casi è consigliabile utilizzare lo strip CON5. In questo caso il segnale di clock sarà applicato al pin 26, che dovrà quindi essere assegnato ad una opportuna primitiva di input.

Nel caso che si utilizzi un segnale di clock esterno, è sempre consigliabile

interporre una resistenza da 200÷300 Ω tra il dispositivo esterno e la CPLD, per prevenire possibili guai. Per le applicazioni più spinte, è sicuramente più pratico utilizzare un oscillatore integrato, come quello montato sulla scheda in figura 4/b; tale circuito utilizzata un oscillatore in DIL8 montato su zoccolo, così da poter utilizzare oscillatori diversi, ed è disponibile a richiesta congiuntamente alla MiniMAX-44.

UN PROGETTO BANALE

Dove si spiega come assegnare i pin di I/O e come programmare una CPLD con MAX II Plus…

Per inaugurare la scheda di sviluppo realizzeremo un semplicissimo progetto, attraverso il quale sarà possibile illustrare le ulteriori funzionalità di base di MAX II Plus. Per creare un nuovo schema, selezionate la voce NEW (menu File), quindi, come abbiamo visto dettagliatamente nella puntata precedente, scegliete il tipo Graphic Editor File e attivate il comando Set Project To Current File (shortcut CTRL+SHIFT+J); quindi salvate il file con il nome di trivial.gdf. Inserite nel foglio di lavoro una primitiva INPUT e una primitiva OUTPUT, collegate l'una all'altra e assegnate il nomi di **clock_in** alla prima e **clock_out** alla seconda. Scegliete un dispositivo attraverso la voce Device (menu Assign) in funzione della CPLD montata sulla MiniMAX-44, ad esempio: EPM3032ALC44–10 e, fatto questo, cliccate con il pulsante destro del mouse sulla primitiva INPUT; nel pop menu che si apre, scegliete la voce Assign\Pin/Location/Chip, come visibile in figura 5.

Figura 5: *Il progetto trivial.gdf*

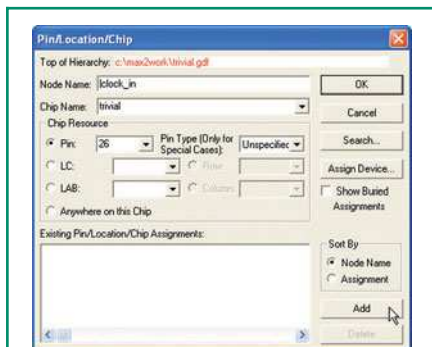



Figura 6: *La finestra Assign\Pin/Location/Chip*

La finestra che si apre (figura 6) consente di assegnare i segnali del vostro progetto ai pin fisici di I/O della CPLD. Seguendo la figura 6, assegnate il pin **26** al segnale **clock_in** e premete il bottone ADD, quindi il pin **16** al segnale **clock_out** e premente di nuovo ADD. È anche possibile cambiare un assegnazione già fatta, selezionando la riga corrispondente, inserendo il nuovo numero del pin e premendo il bottone CHANGE.

Come avrete intuito, questo banalissimo progetto utilizza la CPLD come semplice buffer tra il segnale di clock ed il diodo led DL1, che appunto è connesso sul pin 16. Tornando sull'area di lavoro, assicuratevi che la voce Show Pin/Location/Chip (menu Option) sia spuntata; in questo modo potrete vedere sullo schematico l'assegnazione dei pin: trivial@26 e trivial@16 (figura 7).

trivial@26 clock_in INPUT VCC — OUTPUT clock_out trivial@16

Figura 7: *I pin assegnati trivial@26 e trivial@16*

Un'ultima cosa ancora, prima di compilare il progetto: è necessario assegnare alcune regole generali di sintesi per la compilazione. Scegliete quindi la voce Global Project Logic Synthesis (Menu Assign) e spuntate le varie voci come in figura 8, soprattutto quella relativa ai segnali globali (Automatic Global), che come ho ormai detto più volte, devono essere usati solo da mani esperte. Consiglio caldamente di utilizzare sempre queste impostazioni, perlomeno fino a che non avrete acquisito sufficiente dimestichezza con i dispositivi ed il software. Bene, salvate e compilate il progetto.

Durante la compilazione, MAX II Plus genera diversi file oltre a quello per la programmazione della CPLD. E' spesso utile dare un'occhiata al

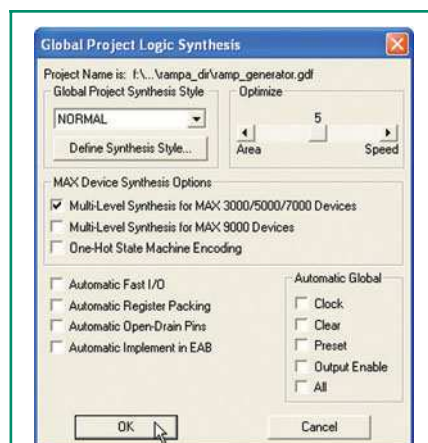



Figura 8: *La finestra Global Project Logic Synthesis*

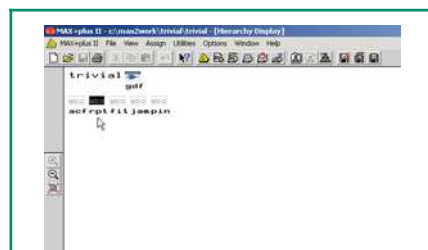



Figura 9: *La finestra Hierarchy Display*

Figura 10: Il pinout della CPLD nel file di report

Figura 11: La finestra Programmer

file di report, che termina con estensione .rpt e quindi, in questo caso, si chiamerà trivial.rpt. Per visualizzare il file, scegliete la voce Hierarchy Display (menu MAX+Plus II) oppure premete sul pulsante della Tool Bar in cui è rappresentata un piramide gialla; come si vede in figura 9 la gerarchia del progetto comprende il file trivial.gdf e tutti i file creati, compreso trivial.rpt.
Se aprite questo file con un doppio click, vedrete che contiene un'infinità di informazioni dettagliate sulle risorse utilizzate e su quelle residue, sugli stili di sintesi utilizzati e, soprattutto, uno schema del **pinout** completo della CPLD dopo la programmazione (figura 10). I pin contrassegnati con la scritta RESERVED sono inutilizzati e vengono connessi a massa internamente alla CPLD, quindi è consigliabile lasciarli non collegati.

Programmare il dispositivo
Scegliete la voce Programmer (menu Max+Plus II), si aprirà la finestra di programmazione (figura 11); sulla destra vengono elencati il nome del file di programmazione, nel nostro caso trivial.pof, il dispositivo scelto ed il valore di checksum. Se non l'avete mai fatto prima, dovete prima di tutto selezionare l'hardware per la programmazione; selezionate quindi la voce Hardware Setup (menu Option) e selezionate Byteblaster(MV), assicurandovi che la porta parallela sia quella a cui avete effettivamente collegato la Byteblaster.

Inserite il connettore JTAG nella MiniMAX-44 e premete il tasto PROGRAM. Vedrete un indicatore avanzare tra le funzioni Esamine, Program e Verify e in pochi secondi la vostra CPLD sarà programmata. Come abbiamo avuto modo di precisare nella seconda puntata, durante la programmazione il dispositivo si sconnette dal mondo esterno, collegando internamente al chip dei deboli pull-up ad ogni pin di I/O; al termine della programmazione, tutti i pin di I/O del dispositivo assumono la configurazione voluta e la CPLD comincia immediatamente a funzionare.

Se non l'avete fatto prima, infilate quindi la schedina di clock nel connettore CON5 (oppure collegate un segnale di clock esterno al pin **26**), regolate il trimmer in modo che presenti la massima resistenza e inserite il jumper JP5, in modo da alimentare i diodi led. Vedrete immediatamente lampeggiare il diodo DL8; la MiniMAX-44 con la schedina di clock inserita è visibile in figura 12.
Per sperimentare un po', potete provare ad assegnare un pin di I/O diverso al segnale clock_out, ricompilare e riprogrammare. Oppure, potete passare al prossimo progetto.

Figura 12: Il sistema di sviluppo MiniMAX–44 completo

UNA PISTOLA LASER

Dove si incontra per la prima volta il MegaWizard Plug-In Manager...
Questo secondo progetto utilizza tutti gli 8 led disponibili sulla MiniMAX-44. Si tratta di un semplice gioco di luci, ma attraverso questo circuito conosceremo uno degli strumenti più interessanti di MAX II Plus, il MegaWizard Plug-In Manager. Questo strumento permette di utilizzare una serie di macro, ovvero di blocchi digitali generici, scegliendo tra funzioni logiche aritmetiche, combinatorie e sequenziali; ognuna di queste funzioni è disponibile all'interno di MAX II Plus e può essere configurata secondo le nostre necessità con pochi click di mouse.

Per cominciare create una cartella chiamata lasergun_dir all'interno della vostra area di lavoro (per esempio c:\max2work\lasergun_dir\), quindi, avviate un nuovo progetto, attraverso la voce NEW (menu File); selezionate come al solito la voce Set Project To Current File (shortcut CTRL+SHIFT+J) e salvate il file con il

nome di lasergun.gdf. Adesso fate doppio click sull'area di lavoro e, nella finestra di inserimento simboli, ciccate sul bottone MegaWizard Plug-In Manager.

Nella prima finestra del Wizard, selezionate "Create a new custom megafunction variation" e premete OK. Nella finestra successiva (figura 13) sono elencati sulla sinistra i blocchi logici disponibili, raggruppati per tipologia; nel nostro caso vogliamo configurare uno shifter register. Quindi scegliete:
STORAGE/LPM_SHIFTREG e inserite il nome:
c:\max2work\lasergun_dir\shift_gun nel riquadro a destra e premete sul bottone NEXT.

Nella finestra successiva, configurate tutte le voci come in figura 14, quindi scegliete sempre OK. Dopo l'ultimo OK, potrete inserire nel foglio di lavoro il simbolo shift_gun (figura 15) che è, appunto, uno shifter register configurato a modo vostro.
Aggiungete quindi due input, il bus q[7..0] e le otto uscite (figura 16); quindi selezionate il vostro target device, ad esempio:
EPM3032ALC44–10 e infine assegnate i pin di I/O come visibile in figura 17 e come elencato nella tabella 6.

Figura 13: *Il MegaWizard Plug-In Manager*

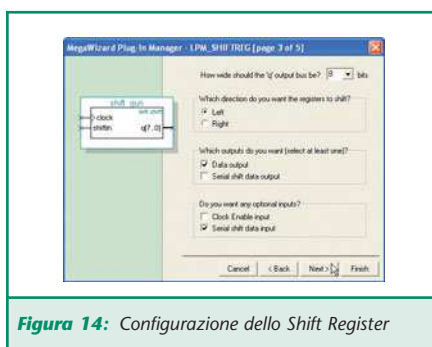



Figura 14: *Configurazione dello Shift Register*

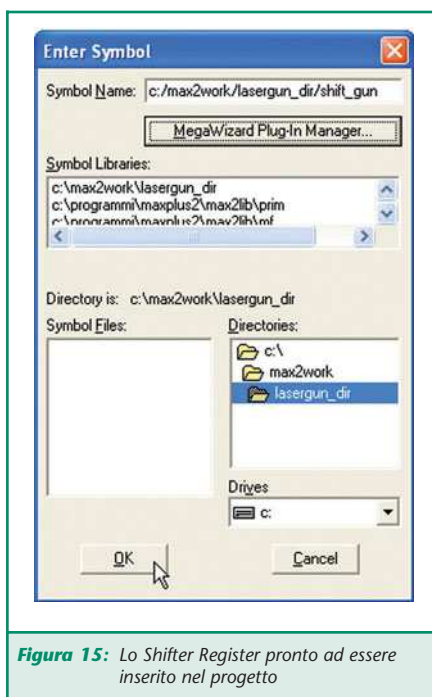



Figura 15: *Lo Shifter Register pronto ad essere inserito nel progetto*

Pin	Assegnazione
clock_in	26
shot	14
q0	41
q1	40
q2	24
q3	21
q4	20
q5	19
q6	18
q7	16

Tabella 6: *Pin assignment per il progetto Lasergun*

Ricordo che per assegnare il nome ad un bus, è sufficiente posizionare il cursore su di esso e poi, cliccando con il pulsante destro, scegliere la voce Enter Node/Bus Name. Dopo aver assegnato le proprietà globali di sintesi come in figura 8, salvate e compilate il progetto. Infine programmate il dispositivo.
Il funzionamento di uno Shifter Register così configurato molto semplice: ad ogni colpo di clock, la configurazione binaria dei registri interni scorre di un posto verso il bit più significativo (q8); nel bit meno significativo (q0) entra di volta in volta il valore presente sul pin **shot**, che è connesso al pulsante **S1**.

Settate la frequenza del clock a 20÷30 Hz e ricordatevi di inserire i jumper JP4 e JP5. In pratica, gli 8 led scorreranno simulando, ad ogni pressione del pulsante S1, il raggio di una pistola laser. Bè, più o meno…

Figura 16: *Il progetto lasergun*

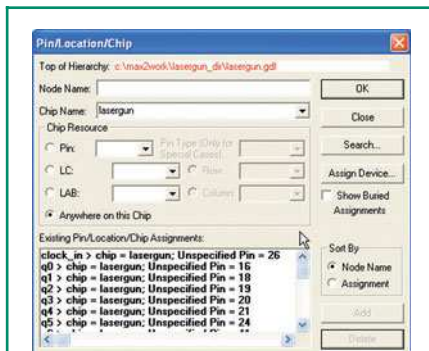



Figura 17: La finestra Assign per il progetto lasergun

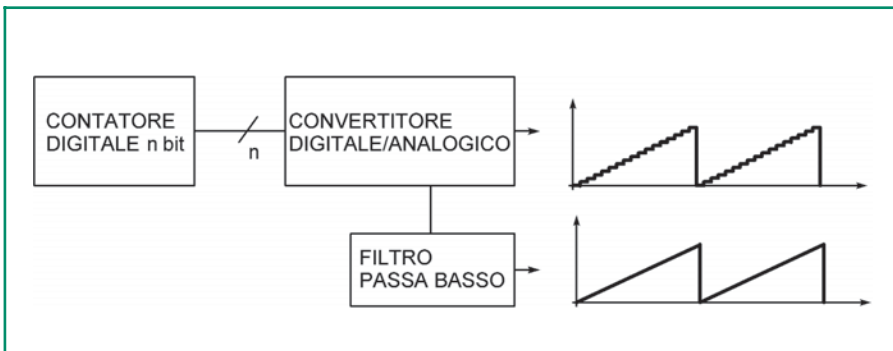



Figura 18: Schema a blocchi di un generatore digitale di rampa

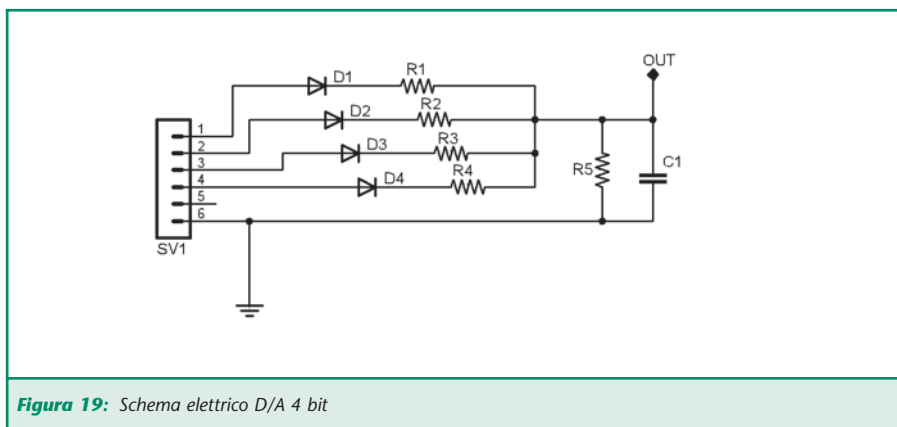



Figura 19: Schema elettrico D/A 4 bit

Sigla	Valore
R1	1 KΩ 1/4 W
R2	2,2 KΩ 1/4 W
R3	4,7 KΩ 1/4 W
R4	8,2 KΩ 1/4 W
R5	330 Ω 1/4 W
D1÷D4	1N4148
C1	0,1 μF poliestere
CON1	Connettore strip 90° M 6 pin

Tabella 7: Lista componenti scheda D-A

UN GENERATORE DI RAMPA

Dove si comincia fare sul serio e si parla apertamente di mixed-signal…

Questa applicazione costituisce un esempio di interfacciamento tra un applicazione prettamente digitale come quella costituita da una logica programmabile ed il mondo analogico; un progetto, insomma, mixed-signal…

L'idea è di realizzare un semplice generatore di rampa, oppure, inserendo un opportuno filtro, un segnale a dente di sega. Lo schema a blocchi del dispositivo è riportato in figura 18; il primo blocco digitale è un contatore binario a n bit, il secondo blocco è un convertitore D/A, sempre a n bit. IL terzo blocco è un filtro passa basso analogico, che può essere anche costituito da una semplice rete RC.

Per la nostra applicazione, utilizzeremo un contatore a 4 bit e una schedina D/A esterna costruita da una semplice rete resistiva; i risultati non saranno eccellenti ma nessuno vi impedisce, successivamente, di aumentare la risoluzione del contatore e utilizzare un D/A integrato. Per questo tipo di schema, le resistenze devono avere valore crescente in maniera proporzionale a 2^n ed è importante che R5 sia almeno 3 volte più piccola di R1.

Per cominciare create una cartella chiamata ramp_dir all'interno della vostra area di lavoro (per esempio: c:\max2work\ramp_dir\), quindi,

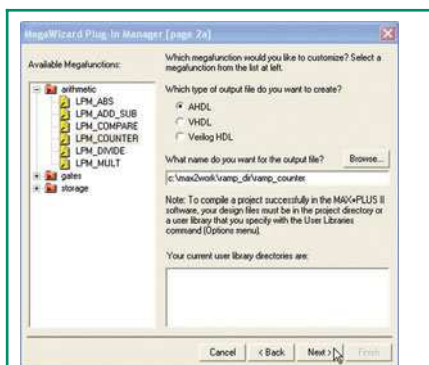



Figura 20: Il MegaWizard per il progetto ramp_generator

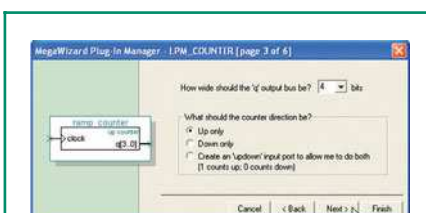



Figura 21: Configurazione del Contatore

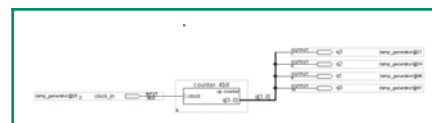



Figura 22: Il progetto ramp_generator

avviate un nuovo progetto, attraverso la voce NEW (menu File); selezionate come al solito la voce *Set Project To Current File* (shortcut CTRL+SHIFT+J) e salvate il file con il nome di *ramp_generator.gdf*. Adesso fate doppio click sull'area di lavoro e, nella finestra di inserimento simboli, ciccate sul bottone MegaWizard Plug-In Manager.

Nella prima finestra, selezionate "Create a new custom megafunction variation" e premete OK. In questo caso vogliamo configurare un contatore, quindi scegliete ARITHMETIC/LPM_COUNTER, inserite il nome:
c:\max2work\ramp_dir\ramp_cnt
nel riquadro a destra relativo al nome e premete sul bottone NEXT (figura 20).
Inserite i parametri nella finestra successiva come in figura 21, struttura a 4 bit e conteggio "up only"; quindi premete sempre OK nelle finestre successive. A questo punto potrete inserire nel foglio di lavoro il contatore configurato a modo vostro; aggiungete quindi sul foglio dello schematico una primitiva **input** per il clock, disegnate il bus **q[3..0]** e le quattro uscite come visibile in figura 22; assegnate il device, per esempio EPM3032ALC44-10 e quindi i pin di I/O come nella tabella 8.

Dopo aver assegnato le proprietà globali di sintesi come in figura 8, salvate e compilate il progetto; quindi, dopo aver inserito la schedina D/A sul connettore CON3, facendo attenzione a che il pin gnd sia inserito al posto giusto, programmate il dispositivo. A questo punto potrete visualizzare sullo schermo dell'oscilloscopio una forma d'onda simile a quella di figura 24.

Pin	Assegnazione
clock_in	26
q0	41
q1	40
q2	24
q3	21

Tabella 8: *Assegnazione pin per il progetto ramp_generator*

Faccio notare che il basso valore di R4 limita l'ampiezza massima a circa 0,5 V e che è possibile utilizzare direttamente il segnale generato solo collegando carichi ad alta impedenza.
Tuttavia, questo semplice progetto illustra con quanta semplicità è possibile utilizzare uno strumento puramente digitale come una CPLD per generare segnali analogici; questo campo applicativo ha avuto uno forte sviluppo e l'utilizzo di tecniche digitali per la sintesi di frequenze e forme d'onda arbitrarie (DDS Digital Direct Synthesis) costituisce un'alternativa molto valida rispetto ai sistemi a PLL.

Probabilmente avremo modo di affrontare l'argomento con maggiore dettaglio, sia da un punto di vista teorico che pratico, in uno dei prossimi numeri della rivista.

DOWNLOAD

I file relativi ai progetti lasergun e ramp_generator presentati in questo articolo sono scaricabili dal sito di fare elettronica.

DOVE REPERIRE IL MATERIALE

Ho acquistato la scheda di sviluppo MiniMAX-44 presso la ditta Digitronix (www.digitronix.it), ad un prezzo di 49 €. L'indirizzo completo è:

Digitronix
Via di Selvamaggio, 21/a
53032 Colle di Val d'Elsa - SIENA
Tel. 0577 920030
digitronix@digitronix.it

Sul sito troverete tutti gli ulteriori dettagli relativi all'acquisto e alle spese di spedizione, il prezzo degli accessori come la schedina di clock, l'alimentatore e le CPLD MAX30xx e MAX70xx.

NELLA PROSSIMA PUNTATA

Nonostante le promesse, ragioni di spazio ci costringono a rimandare alla prossima puntata la parte riservata ai linguaggi di descrizione dell' hardware tipo VHDL e AHDL, ma vi assicuriamo che l'attesa verrà premiata: realizzeremo infatti con pochi click di mouse un generatore PWM e un controller per motori passo passo unipolari, provvisto di pin di direzione e selezione half/full step.

Vi anticipiamo che una MAX3032 da pochi Euro può ospitare comodamente diversi blocchi logici di questo tipo, per sottolineare quanto l'approccio programmabile possa costituire un'alternativa anche economicamente vantaggiosa alle soluzioni standard.

Arrivederci a presto.

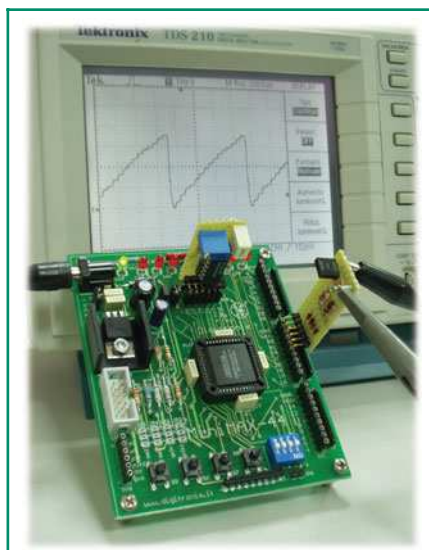



Figura 23: *Il segnale di rampa all'uscita del D/A*

FIERA dell'A.B.C. ELETTRONICA

3/4 aprile 2004 · ERBA (Como)

ore 9/18

hardware · surplus · elettronica · software · foto cine video · componenti · accessori · radiantismo · hobbistica · radio e disco d'epoca · ricezione satellitare · astronomia · telefonia · videogiochi

LARIO FIERE
Viale Resegone

mostra mercato

Expo Elettronica
Il grande Circuito di Fiere dell'elettronica & Co.

organizzazione
BLU NAUTILUS srl
tel. 0541 439573
www.blunautilus.it

Rif. FARE ELETTRONICA

NEWS ON LINE!

servizio gratuito!!
Per essere sempre aggiornati sulle fiere di elettronica inviate la vostra e-mail a info@exporadioelettronica.it, scrivendo nell'oggetto "iscrivimi news on line"

Per ottenere un INGRESSO RIDOTTO scarica il biglietto dal sito www.blunautilus.it o presenta questa inserzione alla cassa

PIC® MICROCONTROLLER BY EXAMPLE

quinta parte

di ***Tiziano Galizia*** *(t.galizia@farelettronica.com)*
e ***Sergio Tanzilli*** *(s.tanzilli@farelettronica.com)*

Nella scorsa puntata abbiamo terminato, almeno per un po', la parte teorica. Torniamo finalmente a realizzare qualcosa di pratico. Dopo aver realizzato nelle lezioni precedenti dei semplici esperimenti con diodi LED e pulsanti, vediamo ora come interfacciare il nostro PIC16F84 ad un display LCD da 2 linee e 16 caratteri.

In questa puntata vedremo come utilizzare un display a cristalli liquidi o LCD (dall'inglese Liquid Crystal Display) dotato di 2 linee di 16 caratteri ciascuna, controllandolo con il PIC16F84.
I display LCD più comuni reperibili in commercio, dispongono di una un'interfaccia ideata da Hitachi che nel tempo è diventata uno standard industriale utilizzato anche da altre case produttrici.
Questo tipo di interfaccia prevede che il display sia collegato al micro tramite un bus dati da 4 o 8 linee più 3 linee di controllo e le linee di alimentazione. Nella tabella che segue è riportata una breve descrizione delle funzioni svolte dai pin normalmente disponibili su questo tipo di display.
Dallo schema elettrico, possiamo notare come, per ridurre al massimo i collegamenti tra il PIC ed il display LCD, sia stata utilizzata la modalità di collegamento dati a 4 bit, impiegando soltanto le linee DB4, DB5, DB6 e DB7.
Le linee DB0, DB1, DB2 e DB3 non saranno utilizzate ma collegate a massa.
Anche la linea R/W non sarà utilizzata e collegata direttamente a massa, con il risultato di aver selezionato la modalità di funzionamento in sola scrittura. In pratica potremo solo inviare dati all'LCD ma non riceverli.

HELLO WORLD!

Prima di proseguire, è necessario scaricare dal sito di Fare Elettronica (www.farelettronica.com) il programma d'esempio LCD1.ASM con cui dovremo programmare il PIC per realizzare il nostro esercizio. Per ragioni di spazio non abbiamo potuto inserirlo in questo articolo (sono circa 6 pagine).
Una volta montato il tutto sulla nostra classica "millefori", ricontrolliamo il montaggio per sicurezza e diamo alimentazione...
Se tutto è andato per il meglio, all'accensione del circuito apparirà sul display la schermata riportata di seguito, ovvero "HELLO WORD !" che è diventata ormai la frase di iniziazione che tutti gli aspiranti programmatori devono aver visualizzato almeno una volta nella vita. Non potevamo essere certo noi ad eludere questa tradizione.

```
HELLO WORLD!_
```

Il risultato non è molto esaltante, ma la sostanza di quello che siamo riusciti a fare è notevole e va studiata approfonditamente.
Se non riuscite a visualizzare nulla sul display, nonostante siate più che sicuri di aver realizzato il circuito correttamente e che il PIC sia stato ben programmato (ricordatevi a proposito di programmare l'oscillatore in modalità XT e di disabilitare il Watch Dog Timer), sarà forse necessario regolare il contrasto del display LCD agendo sul trimmer R2 connesso al pin 3 del display.

LE LINEE ENABLE (E) E REGISTER SELECT (RS) DELL'LCD

Per poter visualizzare una scritta sul display, il PIC deve inviare tutta una serie di comandi tramite le linee del bus dati (linee da DB4 a DB7). Per far questo utilizza due linee di controllo con cui comunica al display l'operazione di trasferimento che cerca di compiere sul bus. Le due linee di controllo sono la linea Enable (pin 6 dell'LCD) e Register Select (pin 4 dell'LCD).
Con la linea Register Select (RS) il PIC segnala al display che il dato presente sul bus è un comando (RS=0) o un dato da visualizzare (RS=1).

Pin	Nome	Funzione
1	GND	GROUND - Questo pin deve essere collegato al negativo di alimentazione
2	VDD	POWER SUPPLY - Questo pin deve essere collegato al positivo di alimentazione (+5 V)
3	LCD	LIQUID CRYSTAL DRIVING VOLTAGE - A questo pin deve essere applicata una tensione variabile da 0 a 5 V tramite un trimmer per regolare il contrasto
4	RS	REGISTER SELECT - Questo pin è un'altra linea di controllo con cui si comunica al display se si sta inviando sul bus dati (linee da DB0 a DB7) un comando (RS = 0) oppure un dato (RS = 1)
5	R/W	READ/WRITE - Questo pin è un'altra linea di controllo con cui si comunica al display se s iintende inviare (R/W = 0) oppure leggere (R/W = 1) un dato
6	E	ENABLE - Questo pin è un'altra linea di controllo con cui si può abilitare il display ad accettare dati ed istruzioni dal bus dati (E = 1)
7	DB0	DATA BUS LINE 0 - Linea dati bit 0
8	DB1	DATA BUS LINE 1 - Linea dati bit 1
9	DB2	DATA BUS LINE 2 - Linea dati bit 2
10	DB3	DATA BUS LINE 3 - Linea dati bit 3
11	DB4	DATA BUS LINE 4 - Linea dati bit 4
12	DB5	DATA BUS LINE 5 - Linea dati bit 5
13	DB6	DATA BUS LINE 6 - Linea dati bit 6
14	DB7	DATA BUS LINE 7 - Linea dati bit 7

Tramite i comandi il PIC può segnalare al display il tipo di operazione da compiere, come ad esempio spostare il cursore o pulire lo schermo.
Con i dati il PIC può inviare al display direttamente i caratteri ASCII da visualizzare.
La linea Enable abilita il display a leggere il comando o il dato inviato sul bus dal PIC.
Il PIC deve preoccuparsi di aver già inviato sul bus dati il comando o il dato giusto prima di mettere a 1 il segnale di enable.

MULTIPLEX SUL BUS DATI

Sia i comandi che i dati sono rappresentati da numeri a 8 bit, quindi come è possibile inviarli al display se il bus dati è composto da sole 4 linee?
Viene fatta in pratica una operazione detta di "multiplex", ovvero ogni byte viene scomposto in due gruppi di 4 bit i quali sono quindi trasmessi sul bus dati in sequenza. Vengono inviati prima i quattro bit meno significativi seguiti dai quattro bit più significativi.
Nel nostro sorgente di esempio, tutte le operazioni di trasmissione dati e comandi verso il display sono eseguite da una serie di subroutine, semplificando così la complessità del nostro programma.

Prima di addentrarci nello studio delle singole subroutine, vediamo come funziona il programma principale.

IL SORGENTE LCD1.ASM

Nella prima parte del nostro sorgente sono definite alcune costanti:

```
;Linee di controllo LCD
LCD_RS    equ 2 ;RS
LCD_E     equ 3 ;E

;Bus dati LCD
LCD_DB4   equ 4 ;DB4
LCD_DB5   equ 5 ;DB5
LCD_DB6   equ 6 ;DB6
LCD_DB7   equ 7 ;DB7
```

Queste costanti definiscono l'associazione tra le linee del PIC (tutte connesse alla PORTA B) e le linee del display. Le singole definizioni saranno usate all'interno delle subroutine di gestione dell'LCD al posto dei singoli numeri di identificazione delle linee di I/O.

```
tmpLcdRegister res 2
msDelayCounter res 2
```

Di seguito viene allocato spazio per due registri: *tmpLcdRegister*, usato dalle routine di gestione dell'LCD e *msDelayCounter* usato dalla subroutine *msDelay* che genera dei ritardi software da 1ms per il contenuto del registro W. Questa subroutine viene utilizzata sempre dalle subroutine di gestione per generare le temporizzazioni richieste durante la trasmissione di dati e comandi all'LCD.

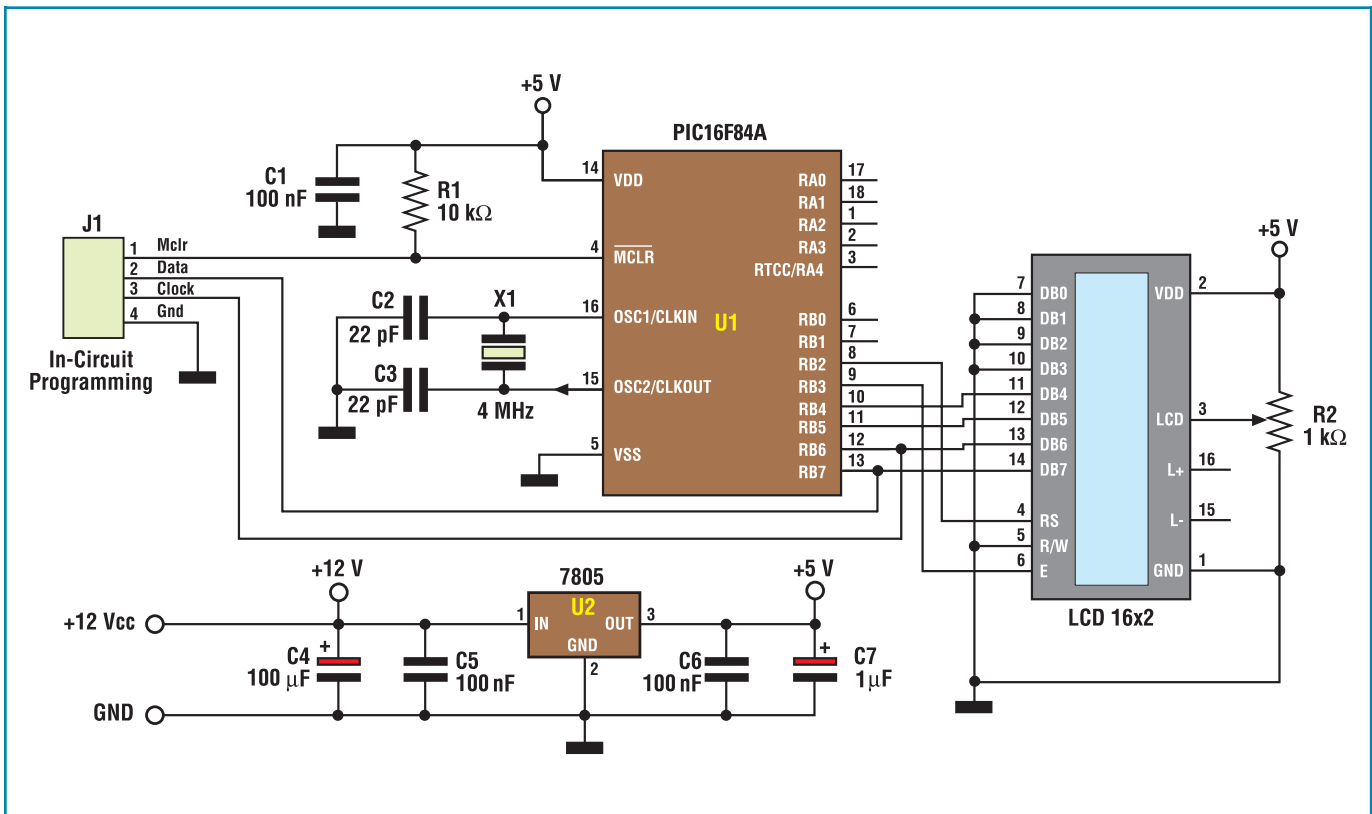



Figura 1: Schema elettrico dell'interfaccia a display LCD

Segue una parte di definizione delle linee di connessione tra il PIC ed il display e quindi arriva la prima chiamata a subroutine che ci interessa.

```
call    LcdInit
```

LcdInit è una subroutine che deve essere chiamata solo una volta all'inizio del programma e prima di qualsiasi altra subroutine di gestione dell'LCD. Essa si occupa di effettuare tutte le operazioni necessarie per inizializzare correttamente l'LCD e consentire, alle funzioni successive, di poter operare correttamente.
Con le istruzioni seguenti:

```
movlw   00H
call    LcdLocate
```

si posiziona il cursore del display sulla prima riga e prima colonna dello schermo. I caratteri inviati successivamente saranno visualizzati a partire da questa posizione. I quattro bit più significativi del valore caricato nel registro W, con l'istruzione *movlw 00H*, contengono il numero della riga dove si vuole posizionare il cursore, mentre i quattro bit meno significativi contengono il numero della colonna. Provando a cambiare il valore nel registro W possiamo ottenere posizionamenti diversi. Con il valore10H, ad esempio, otterremo la schermata seguente:

A questo punto, per visualizzare ogni carattere della scritta vengono utilizzate le seguenti istruzioni:

```
movlw   'H'
call    LcdSendData
```

e così via per ogni lettera da visualizzare. Da notare che l'incremento della posizione del cursore avviene automaticamente.

SUBROUTINE DI GESTIONE DEL DISPLAY LCD

Vediamo brevemente quali funzioni svolgono le subroutine di gestione del display LCD, fornite con il sorgente LCD1.ASM.

LcdInit

Questa subroutine si occupa di inizializzare il display LCD e di pulire lo schermo. Deve essere chiamata una sola volta e prima di qualsiasi altra subroutine di gestione dell'LCD.
Non richiede alcun passaggio di parametri.

LcdClear

Pulisce il contenuto dello schermo e riposiziona il cursore sulla prima colonna della prima riga.

Non richiede alcun passaggio di parametri.

LcdLocate
Serve a posizionare arbitrariamente il cursore all'interno dell'area visibile del display.
Richiede il valore di riga e colonna, per il posizionamento del cursore, nel registro W. I bit da D0 a D3 contengono il valore di colonna (asse Y) mentre i bit da D4 a D7 il valore di riga (asse X). La numerazione delle righe parte da zero in alto. La numerazione delle colonne parte da 0 a sinistra.

LcdSendData
Serve ad inviare un carattere ASCII all'LCD, da visualizzare nella posizione corrente del cursore.
Richiede nel registro W il valore ASCII del carattere da visualizzare.

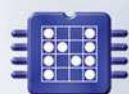
LcdSendCommand
Serve ad inviare un comando all'LCD. I comandi riconosciuti dall'LCD sono riportati sul datasheet dello stesso.
Richiede nel registro W il valore ad 8 bit del comando da inviare.

LcdSendByte
Questa funzione viene utilizzata internamente dalle altre funzioni di gestione dell'LCD e si occupa di effettuare lo split di dati e comandi ad 8 bit sul bus dati a 4 bit.

CONCLUSIONI

Maggiori informazioni sul funzionamento degli LCD possono essere reperite direttamente sui datasheet che il rivenditore di fiducia dovrebbe essere in grado di fornire.
Le subroutine presentate in questa lezione dovrebbero poter funzionare, con modifiche minime o nulle, anche su LCD di diverso taglio.

Vi ricordiamo che è possibile scaricare il codice sorgente (LCD1.ASM) dell'esperimento proposto, dal sito di Fare Elettronica: www.farelettronica.com

Riprenderemo sul prossimo numero con un altro esercizio in cui amplieremo le funzioni di questo circuito con una porta seriale RS232 che collegheremo al nostro PC.

PRATICAMENTE... INTERFACCIAMOCI CON LE PORTE LOGICHE (INTERRUTTORE TEMPORIZZATO)

di Maurizio Del Corso
m.delcorso@farelettronica.com

Puntuali come sempre, anche questo mese prendiamo la nostra pillola di elettronica. In questo numero approfondiremo lo studio dei circuiti tipici usati per applicare i segnali d'ingresso ai circuiti logici e per prelevarne le uscite pilotando carichi che altrimenti non potrebbero essere comandati. Come applicazione di esempio realizzeremo un interruttore temporizzato ovvero un circuito che in seguito alla pressione di un pulsante, accende una lampada per circa 1 minuto.

ANALISI E SPECIFICHE DEL PROBLEMA

Si deve progettare un circuito con un ingresso ed un'uscita in cui premendo un pulsante si accende una lampada da 12 V / 5 W per circa 1 minuto. Il sistema può essere suddiviso in due blocchi (figura 1): il primo blocco riceve il comando d'ingresso e genera in uscita un impulso della durata di circa 1 minuto; il secondo blocco ha la funzione di azionare la lampada per la stessa durata dell'impulso d'ingresso.

LA SOLUZIONE OTTIMALE

Tra le possibili soluzioni, in figura 2 è riportata quella più economica e "didatticamente interessante". Si riconoscono il circuito d'ingresso (costituito dal pulsante e dal resistore R1), il generatore d'impulsi (costituito dalle due porte logiche, dal condensatore e dalla resistenza R) ed il circuito di pilotaggio della lampada costituito dal transistor, R2, il diodo ed il relé.

Iniziamo l'analisi dello schema dal circuito d'ingresso.
Quando il pulsante non è premuto, nella resistenza R1 non scorre corrente quindi l'ingresso della porta logica si trova a massa, dunque a livello logico "0".
Premendo il pulsante, l'ingresso

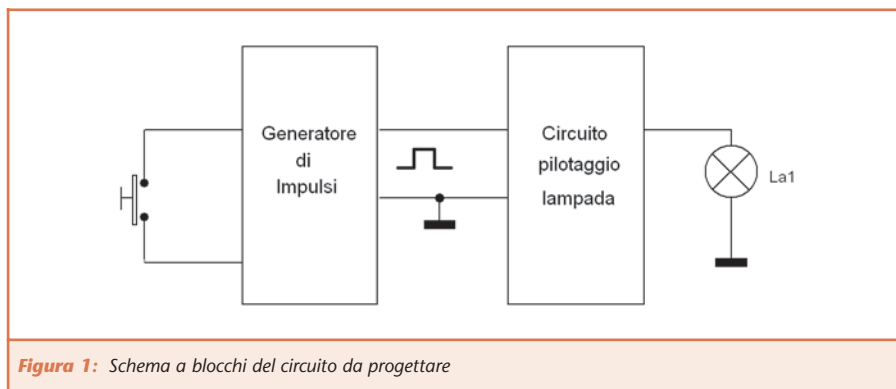



Figura 1: *Schema a blocchi del circuito da progettare*

della porta logica viene collegato al positivo di alimentazione (+5V) quindi portato al livello logico alto. In questo caso è richiesta dall'alimentazione una corrente pari a Vcc/R1 dove Vcc è il valore della tensione di alimentazione (nel nostro caso +5V).
Con i valori scelti per il circuito di figura 2, tale corrente è limitata a circa 0,5mA. La resistenza R1 ha quindi una duplice funzione: limita la corrente quando il pulsante è chiuso, e funge da pull-down quando il pulsante è aperto.

Analizziamo ora il blocco "generatore d'impulsi" che in gergo tecnico è chiamato "multivibratore monostabile".
Le porte logiche utilizzate sono NOR di tipo CMOS. La porta NOR fornisce in uscita un livello basso se almeno uno dei suoi ingressi è alto ed in generale le porte CMOS sono caratterizzate da una resistenza d'ingresso molto alta, il che significa che gli ingressi assorbono una corrente praticamente trascurabile.
La porta B ha i due ingressi collegati insieme perciò funziona come porta NOT, vale a dire l'uscita è bassa se l'ingresso è alto e viceversa (vedi tabella della verità nel riquadro di approfondimento).
Per semplicità indicheremo i livelli logici con "0" ed "1" ai quali corrispondono rispettivamente i livelli di tensione 0 V e 5 V.

Supponiamo che l'uscita della porta B sia bassa (V4="0") e che il pulsante non sia premuto (V1="0"): se V4="0" allora V3="1" ed essendo V1="0" si ha che anche V2="1".
Il circuito è quindi in uno stato di equilibrio in cui l'uscita vale zero ed il condensatore è scarico (essendo V2=V3).
Premendo e rilasciando il pulsante sarà applicato un impulso positivo all'ingresso 1 della porta A, quindi, l'uscita di tale porta commuterà verso lo zero logico. Questa brusca variazione, si trasmette inalterata su V3, quindi, l'uscita della porta B commuta verso il livello logico alto. Il condensatore inizia a caricarsi tramite la resistenza R facendo salire la tensione all'ingresso della porta B. Quando tale tensione raggiunge un valore limite, per cui viene riconosciuto come "1" logico dalla porta B (supponiamo che tale valore sia la metà della tensione di alimentazione), la tensione V4 tornerà a livello basso e tutto il circuito tornerà nel suo stato di riposo.
Da questa descrizione dovrebbe apparire chiaro che il tempo T in cui l'uscita V4 permane a livello alto, altro non è che il tempo impiegato dal condensatore per caricarsi fino al livello limite (che abbiamo supposto essere 2,5V).
Matematicamente questo intervallo di tempo è indipendente dalla tensione di alimentazione ma dipende solamente dai valori di R e C ed è dato dalla seguente relazione:

$$T=0.693 \cdot R \cdot C \quad \text{[secondi]}$$

Se vogliamo che la lampada resti accesa per circa 1 minuto, si dovrà scegliere R·C=86.56 [sec], quindi fissando C=1000 μF deve essere R=86.56 KΩ, perciò scegliamo R=82 KΩ che è il valore commerciale più vicino.

Nella figura 3 sono riportate le forme d'onda relative ai diversi punti del circuito nell'ipotesi che il pulsante sia premuto all'istante 0.

Analizziamo ora lo stadio di pilotaggio della lampada. Il transistor

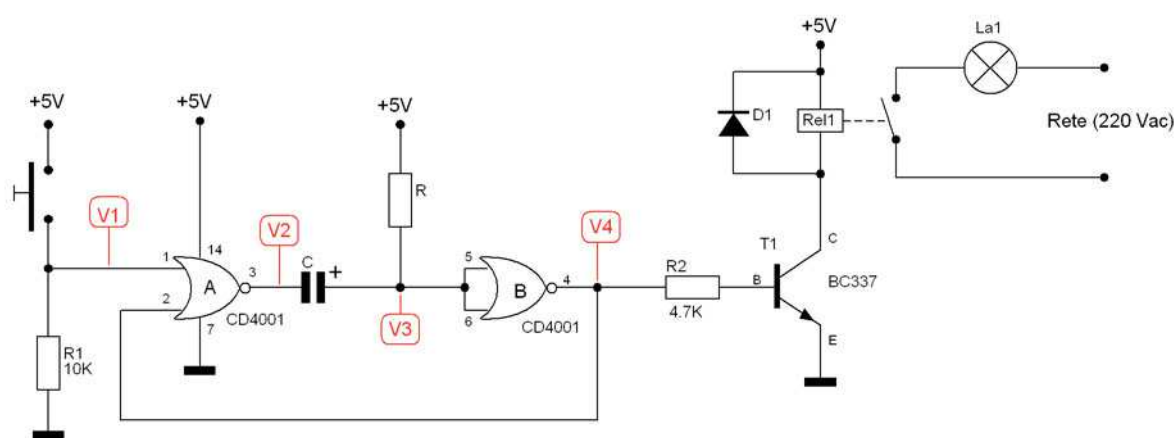
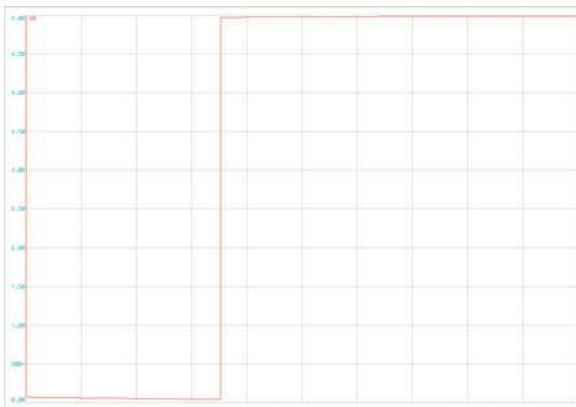
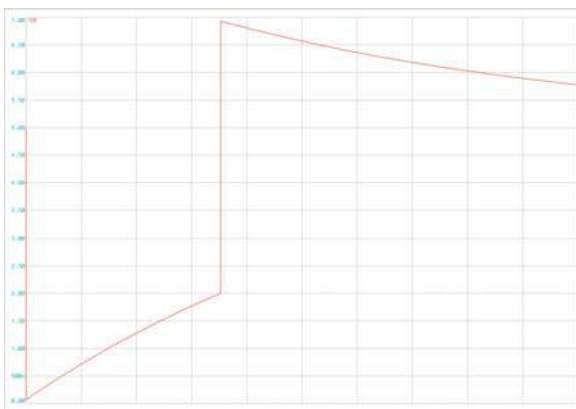



Figura 2: Lo schema per l'interruttore temporizzato

V1

V2

V3

V4

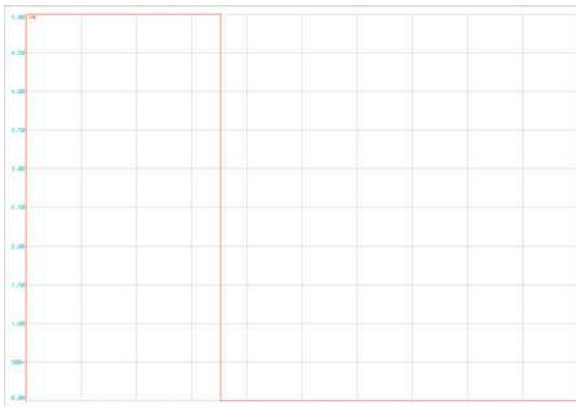



Figura 3: Forme d'onda relative al circuito di figura 2

T1 funziona da interruttore comandato: quando l'uscita della porta B è alta, T1 è in saturazione e la tensione tra collettore ed emettitore scende a circa 0,2 Volt provocando l'eccitazione del relé. Se V4 è a livello basso allora, T1 è interdetto ed il relé non viene eccitato.
La resistenza R2 ha lo scopo di limitare la corrente di uscita della porta, sul livello alto a circa 1mA, valore sufficiente per la saturazione del transistor.

Molto importante è la funzione svolta dal diodo D1. Il relé è costituito da una bobina che svolge la funzione di elettrocalamita quindi presenta una certa induttanza.
Ai capi di una induttanza, si crea una tensione tanto più alta quanto più rapida è la variazione di corrente che percorre l'induttanza stessa.
Quando T1 passa dallo stato di conduzione a quello d'interdizione (equivalente all'apertura di un interruttore) la corrente nella bobina del relé, tende ad annullarsi istantaneamente per questo ai suoi capi si crea una tensione piuttosto alta.
Questa tensione fa sì che il diodo entri in conduzione consentendo alla corrente di continuare a circolare fino ad estinguersi in un

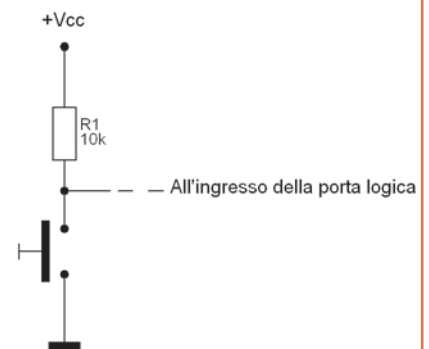



Figura 4: Ingresso basso alla pressione del pulsante

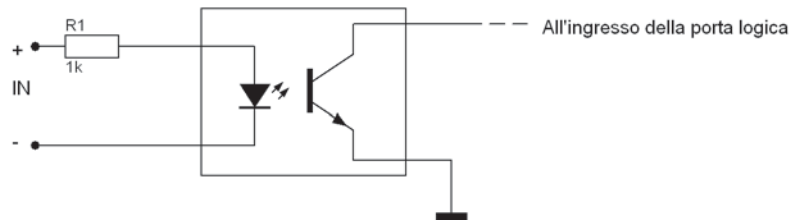



Figura 5: Ingresso optoisolato

Figura 6: Accensione di un led mediante una porta TTL

tempo più lungo, riducendo drasticamente la tensione ai capi della bobina che altrimenti danneggerebbe il transistore T1.

APPROFONDIMENTI

Nell'esempio precedente abbiamo visto come applicare un segnale all'ingresso di una porta logica e come utilizzare la sua uscita per il pilotaggio di un relé. Vediamo ora altre tecniche d'interfacciamento con le porte logiche.

Ingresso basso alla pressione di un pulsante

Se vogliamo fare in modo che alla pressione di un pulsante l'ingresso di una porta logica commuti da livello alto a livello basso, si può utilizzare lo schema di figura 4.
La resistenza in questo caso svolge la funzione di "pull-up" ovvero mantiene alto l'ingresso, quando il pulsante non è premuto. Alla pressione del tasto l'ingresso viene azzerato e la resistenza limita la corrente assorbita dall'alimentazione.

Optoisolamento di un ingresso

L'optoisolamento di un ingresso è necessario quando si vuol garantire l'isolamento galvanico tra il nostro circuito ed il mondo esterno. Lo schema è quello di figura 5.
Attenzione al fatto che il circuito di figura 5 è invertente: applicando un valore alto di tensione il fototransistor va in saturazione quindi alla porta logica giunge un valore basso.

Accensione di un LED

L'uscita delle porte logiche TTL non è in grado di erogare correnti elevate sul livello alto quindi il LED può essere acceso solo sul livello basso. Per fare questo si utilizza il circuito di figura 6.
La resistenza limita la corrente assorbita dalla porta.

Azionamento di un triac

Anziché utilizzare un relé, che è un dispositivo elettromeccanico, è possibile azionare carichi alimentati alla tensione di rete, utilizzando un triac.
Allo scopo si può utilizzare lo schema di figura 7, in cui l'uscita della porta pilota un optotriac che a sua volta pilota il triac.
L'optotriac garantisce l'isolamento galvanico tra la sezione alimentata a bassa tensione e quella alimentata alla tensione di rete.
Utilizzando un triac da 600 V / 5 A è possibile pilotare carichi di 200 W. Per carichi maggiori si dovrà utilizzare un triac diverso collegato ad un dissipatore metallico. La coppia R1, C1 limita i disturbi sulla rete.

Figura 8: Accensione di una serie di LED

Figura 7: Azionamento di un triac

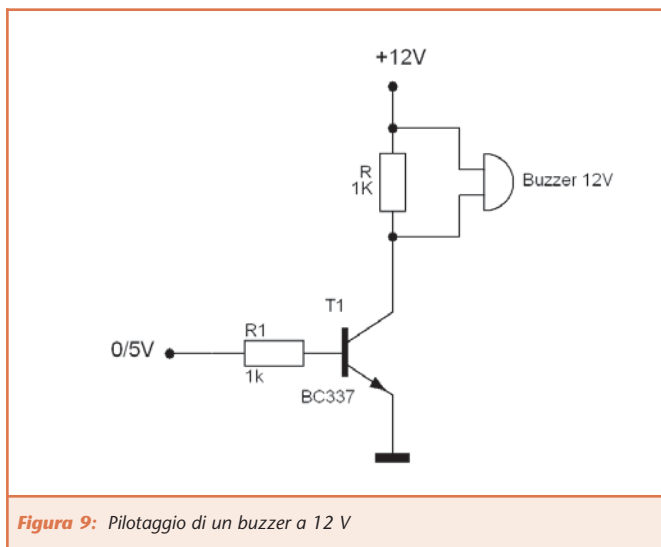



Figura 9: Pilotaggio di un buzzer a 12 V

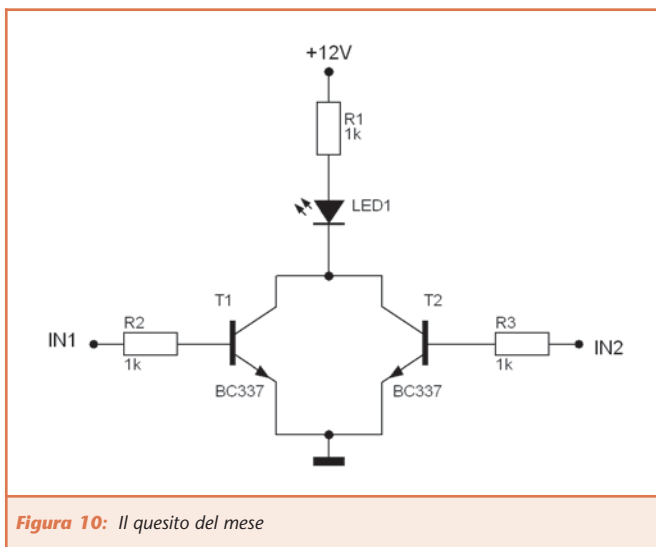



Figura 10: Il quesito del mese

Accensione di una serie di LED
Per accendere simultaneamente una serie di LED è necessario interporre un transistor tra l'uscita della porta ed i LED come mostrato in figura 8. La resistenza R limita la corrente nei LED e dovrà essere di 390 Ω per 6 LED, 470 Ω per 5 LED, 680 Ω per 4 LED, 820 Ω per 3 o 2 LED, 1 KΩ per un solo LED.

Azionamento di un buzzer a 12 V
Spesso per emettere un segnale acustico è necessario pilotare un buzzer. Per questo si può utilizzare lo schema di figura 9, in cui il transistor fornisce la corrente necessaria all'attivazione del buzzer.

IL QUESITO DEL MESE

Questo mese vi propongo il circuito di figura 10: i due terminali potranno essere collegati alle uscite di due porte logiche quindi pilotati con livelli logici di tensione ("0" = 0 V, "1" = 5 V).
In quali condizioni degli ingressi il LED risulterà acceso?
Aspetto le vostre risposte (m.delcorso@farelettronica.com).

Per quanto riguarda il quesito del mese scorso, i due LED del circuito proposto rimarranno spenti.
Questo perché le due batterie polarizzano inversamente entrambi i diodi che risultano così interdetti.

NEL PROSSIMO NUMERO...

Nel prossimo numero ci occuperemo dell'analisi e realizzazione di un alimentatore milleusi con gli stabilizzatori integrati della serie 78xx, 79xx, LM317.

Puntuali, mi raccomando!

CD4001 - 4 porte NOR CMOS

A	B	A+B
0	0	1
0	1	0
1	0	0
1	1	0

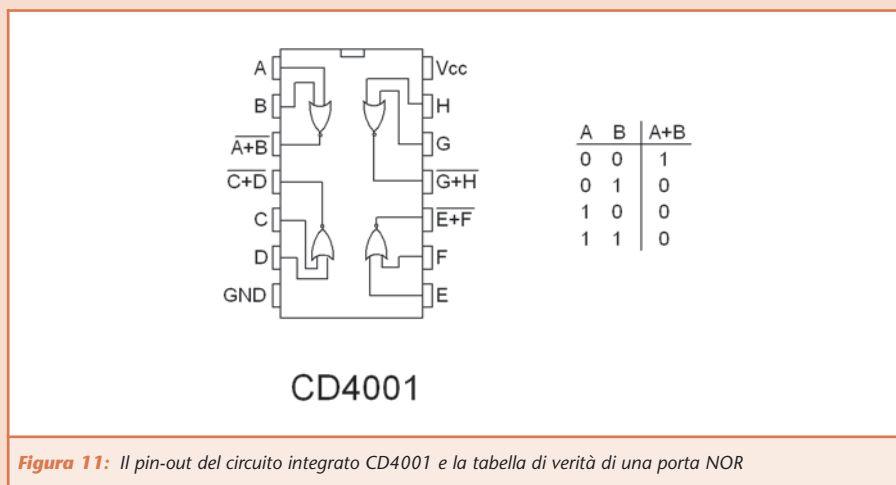
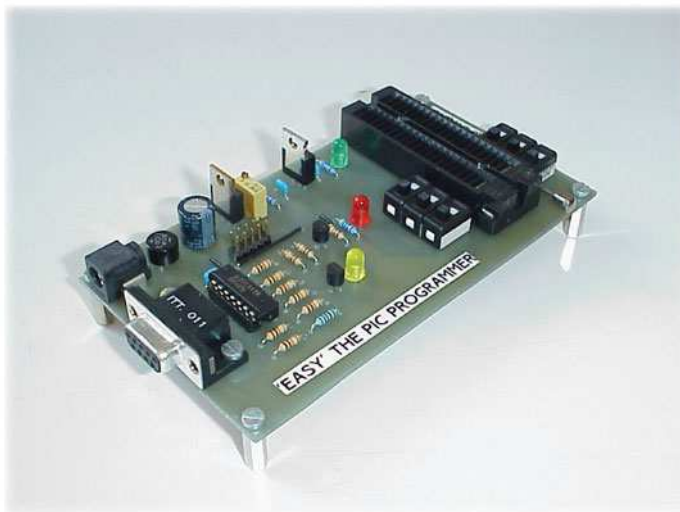



Figura 11: Il pin-out del circuito integrato CD4001 e la tabella di verità di una porta NOR

NOTE SULL'UTILIZZO E LA PROGRAMMAZIONE IN-CIRCUIT CON "EASY", IL PROGRAMMATORE DI PICmicro™

di Fabio Fioravanzo
fabio.fioravanzo@tiscali.it

Nella scorsa puntata vi ho presentato il programmatore di microprocessori PIC "EASY" illustrandone caratteristiche e funzionamento. È ora arrivato il momento di passare alla sperimentazione vera e propria, mettendo in pratica quanto appreso, sfruttando anche quel connettore per la programmazione in-circuit...

Devo dire che il successo di Easy è andato oltre ogni aspettativa!
Personalmente penso di non aver fatto nulla di eccezionale, ma di aver semplicemente reso disponibile un utile strumento a tutti gli hobbisti e sperimentatori elettronici, sia esperti sia principianti, che vogliono lavorare in modo semplice, e soprattutto economico, con i microcontrollori PIC.
Spero anche che, grazie al mio articolo, nuove persone si siano avvicinate al mondo dell'elettronica, e in particolare a quello dei controllori programmabili.
Colgo l'occasione per ringraziare le numerose persone che mi hanno scritto complimentandosi con me per il programmatore da me realizzato, ma anche per chiedere dei chiarimenti, o segnalarmi qualche problema avuto durante la costruzione o l'utilizzo di EASY.

LE FASI DI PROGRAMMAZIONE

Nella scorsa puntata, ci eravamo lasciati dopo aver spiegato come installare e settare IC-PROG. Grazie a questo semplice ed intuitivo software di gestione, abbiamo visto come si possa facilmente, ed in pochi secondi, programmare il "nostro" microcontrollore PIC.
Analizziamo ora più dettagliatamente le varie fasi che ci consentono di programmare il nostro micro PIC con EASY.
Per prima cosa occorre effettuare il collegamento di EASY con la porta parallela del PC, tramite il cavo che vi sarete costruiti secondo quanto indicato nel precedente artico-

lo. Vi ricordo nuovamente di fare molta attenzione nell'effettuare i collegamenti, pena il non funzionamento del programmatore, o peggio la rottura della porta parallela del PC!
Dalla parte del programmatore è anche possibile eliminare il DB9 e saldare direttamente i cavi sul circuito stampato, oppure usare un altro tipo di connettore (ma in questo caso dovreste modificare il circuito stampato). Sta a voi la scelta. Io per maggior praticità ho appunto inserito il connettore DB9.
In figura 1 potete vedere in dettaglio lo schema del cavo da utilizzare per il collegamento.
Effettuato il collegamento, si dà alimentazione al programmatore. Se tutto è OK, il led verde deve accendersi. In caso contrario significa che c'è un problema nella sezione di alimentazione.
Vi rammento, se non lo avete fatto prima, di tarare correttamente il livello della +VPP a 13.5V, tramite il trimmer R1. Infatti una delle principali cause di mancata programmazione dei micro PIC, è proprio il fatto che la +VPP non arriva a 13.5V, ma ad un livello inferiore. Quindi occhio alla taratura!
Si selezionano poi i microswitch a seconda del dispositivo da programmare come indicato in tabella 1.
I PIC vanno inseriti nella parte iniziale dello zoccolo e con la tacca di riferimento verso la levetta dello zoccolo stesso.
Le EEPROM, pur dovendo anch'esse essere inserite con la tacca di riferimento rivolta verso la levetta dello zoccolo, dovranno invece essere posizionate dalla parte opposta rispetto alla levetta.
Si avvia quindi il programma IC-PROG, si seleziona il tipo di dispositivo che si intende programmare e, per i PIC, il tipo di oscillatore, l'eventuale protezione, e i fuses (WDT, MCLR, PWRT, CP, ecc.).

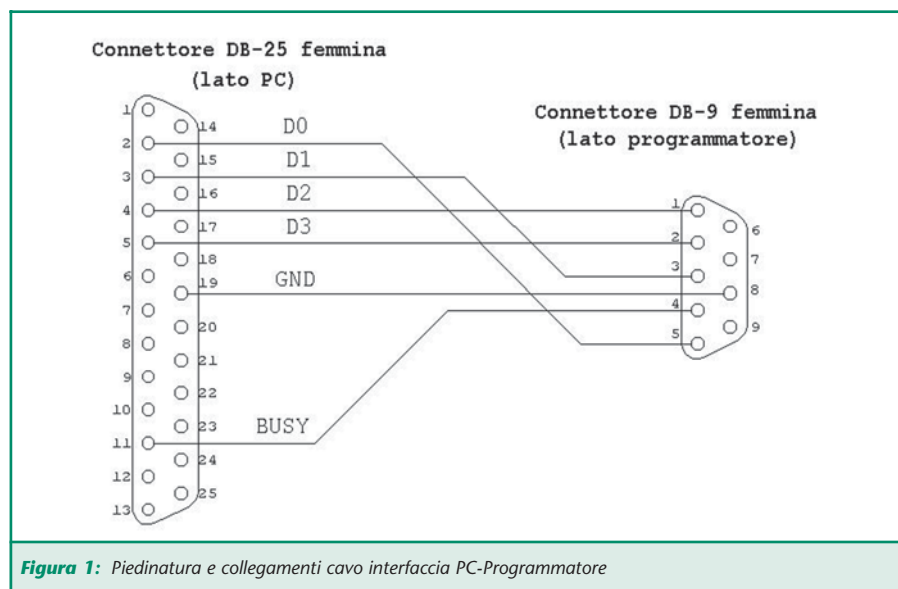



Figura 1: *Piedinatura e collegamenti cavo interfaccia PC-Programmatore*

Se state programmando PIC tipo 16F62X, o 16F87X, occhio a non selezionare LVP (Low Voltage Programming), pena l'apparizione di un errore! Questi dispositivi sono infatti dotati della possibilità di essere programmati senza necessità di far arrivare 13.5V sul MCLR, ma bensì solo i classici 5V. EASY non è compatibile con questi tipo di programmazione, e quindi il relativo fuse non va selezionato. Ad ogni modo fate riferimento al datasheet del vostro dispositivo, per sapere esattamente le funzioni dei vari fuses che lo caratterizzano. Un occhio di riguardo va anche all'eventuale selezione del Code Protection, potreste non riuscire più a leggere o riprogrammare il vostro dispositivo una volta programmato con questa opzione attiva!
Si carica quindi il file HEX con cui si desidera programmare il PIC, cliccando sull'icona <Apri File> e quindi selezionando la cartella e il file voluto.
Ogniqualvolta si inizia una operazione di lettura, scrittura o cancellazione, si accendono i led rosso e giallo, a testimonianza del passaggio dei dati dal PC al PIC.
È sempre meglio prima cancellare il dispositivo e poi avviare la program-

Figura 2: *Messaggio di errore in caso di mancata comunicazione PC-Programmatore*

Dispositivo da programmare	Posizione microswitch
PIC 8/18pin & EEPROM	2
PIC 28/40 pin	1

Tabella 1: *Posizionamento dei microswitch*

mazione.
Se la verifica non dà errori, il vostro dispositivo è perfettamente programmato!
Nel caso apparisse un messaggio come quello di figura 2, significa che ci sono problemi di connessione tra PC e programmatore, oppure il PIC non è stato inserito correttamente nello zoccolo ZIF, o semplicemente, come succede spesso a me, non si è abbassata la levetta…

Un'ultima cosa relativa alla costruzione di EASY; al posto del 74LS07, può essere usato un normale e più

facilmente reperibile 7407, senza per questo creare problemi di sorta. Per quanto riguarda il software se lavorate con Windows XP e la versione 1.05C di IC-PROG, potreste avere dei problemi, vi consiglio quindi di utilizzare la versione 1.05A.

LA PROGRAMMAZIONE IN CIRCUIT

Grazie alla tecnologia Flash dei PIC, possiamo programmarli e riprogrammarli migliaia di volte (per il PIC16F84 il data-sheet indica un numero di scritture della flash pari a 1000 volte prima di non garantirne più il funzionamento). Quindi penso che qualsiasi sperimentatore, prima di aver raggiunto questo limite, riesca a far funzionare il suo software senza bachi!
C'è però un altro problema che a prima vista potrebbe sembrare trascurabile, ma che alla lunga può portare a seri danni, il continuo smontaggio/rimontaggio del micro dal circuito (supponendo di lavorare con un componente DIL zoccolato come la maggior parte degli hobbisti), per riprogrammarlo e poi reinserirlo nel circuito al fine di testare le modifiche fatte, può danneggiare i pin dell'integrato, nonché lo zoccolo stesso. Inoltre tutta l'operazione, se ripetuta più volte, porta ad una grande perdita di tempo. Se poi pensiamo ai microprocessori in versione SMD, che oramai cominciano a diffondersi anche presso i piccoli sperimentatori o presso chi fa piccole serie di prodotti, la programmazione del micro, anche utilizzando il programmatore ufficiale Microchip risulta impossibile, a meno di non acquistare costosi adattatori.
Per fortuna la Microchip ha pensato anche a questo, ed ha dotato tutti i suoi PIC dell'ICSP (In-Circuit Serial Programming) grazie alla quale possiamo programmare il PIC direttamente sulla scheda in cui sarà utilizzato.
Ma andiamo a vedere cosè l'ICSP.
Si tratta di un sistema di programmazione seriale in cui bastano due sole linee per clock e dati, più altre 3 per massa, alimentazione e tensione di programmazione (considerando di alimentare la scheda a parte, i collegamenti programmatore-scheda si riducono a 4).
Per i PIC della serie 16FXX tipicamente le linee per clock e dati sono i pin RB6 e RB7 rispettivamente. In questo modo possiamo montare le nostre schede con micro vergine e programmarle poco prima di venderle o utilizzarle, caricando quindi l'ultima versione del firmware.
Nel realizzare EASY, non ho quindi potuto trascurare questa peculiarità, l'ho quindi dotato di un connettore aggiuntivo a 5+5 vie in cui sono riportati tutti i segnali necessari alla programmazione in-circuit.
Il circuito dove si trova il PIC da programmare dovrà essere progettato ad hoc, come vedremo di seguito, e dotato di un apposito connettore atto a ricevere i segnali dal programmatore. I particolari del connettore e del cavo di collegamento da realizzare, sono riportati in figura 3. Vi ricordo che è preferibile non utilizzare l'alimentazione a 5V, presente sul connettore ICSP, per alimentare la scheda da programmare, se non nel caso in cui la potenza assorbita sia esigua (qualche Watts). Le fasi di programmazione sono le stesse analizzate precedentemente. Naturalmente in questo caso la posizione dei microswitch è ininfluente.

CIRCUITO APPLICATIVO

Se vogliamo che il nostro circuito sia compatibile con l'ICSP, dobbiamo far sì che tutti i segnali di programmazione siano connessi direttamente al PIC.
In figura 4 vediamo un tipico circuito di interfaccia per la programmazione in-circuit. I parametri da considerare per una corretta realizzazione del circuito sono:

- Isolamento del pin MCLR dal resto del circuito.
- Impedenza di carico sui pin RB6, RB7.
- Capacità singola dei pin MCLR, VDD, RB6, RB7.
- Interfaccia verso il programmatore.

Normalmente il pin MCLR si collega ad un circuito RC, in questo caso con interposto un diodo. Il diodo

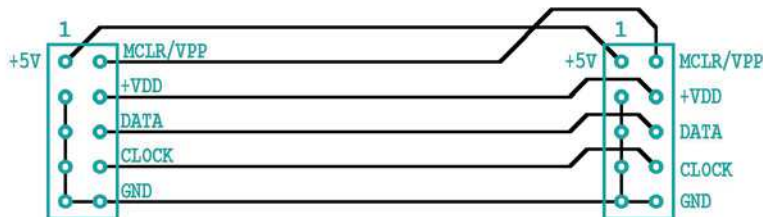



Figura 3: *Piedinatura e collegamenti cavo interfaccia ICSP*

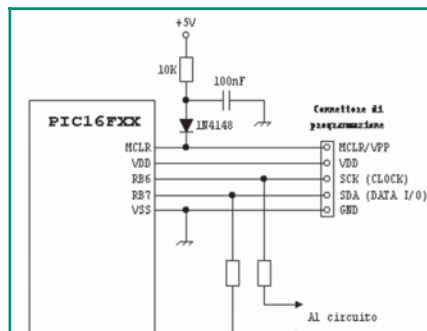



Figura 4: *Circuito esemplificativo per l'adattamento all'ICSP*

dovrebbe essere uno Schottky, ma per i nostri utilizzi un normale 1N4148 va più che bene. L'isolamento del MCLR dal resto del circuito si spiega con il fatto che, durante la fase di programmazione, viene portato a 13.5V e quindi, se fosse direttamente collegato al circuito, potrebbe portare alla rottura di qualche componente.
Anche per RB6 ed RB7, essendo i pin dedicati alla programmazione, sarebbe auspicabile fossero isolati dal resto del circuito, cioè fossero dedicati solo a questa specifica funzione. Se ciò non è possibile, occorre valutare attentamente l'impedenza del circuito.
In figura 4 l'isolamento è fatto con due semplici resistenze, ma ciò non è valido per tutte le applicazioni. Bisogna quindi valutare, di volta in volta, quale sia la soluzione circuitale migliore. Se proprio si debbono usare RB6, RB7, consiglio di usarli come uscite, con una impedenza di carico minima.
Anche l'impedenza del programmatore ha la sua influenza. Eccovi quindi spiegato la presenza dei buffer sulla scheda di EASY.
Non sono da trascurare neppure i condensatori di filtro collegati ai pin del micro dedicati alla programmazione, se la capacità è troppo elevata si rischia di rallentare il flusso delle informazioni e di conseguenza impedirne la programmazione.

Ad ogni modo, per spiegazioni più dettagliate sulla programmazione in-circuit vi rimando all'articolo di Mariano Paolizzi pubblicato sul numero di Dicembre 2003, o ancora alla vasta documentazione presente sul sito Microchip, in particolare il "Mid-Range Reference Manual" (DS33023/A), in cui sono spiegate dettagliatamente tutte le caratteristiche tecniche dei micro PIC.

Beh, penso di aver descritto tutte le caratteristiche di EASY, sia per quanto riguarda il suo utilizzo, che per quanto riguarda la programmazione in-circuit.
Nella prossima puntata presenterò una scheda per esperimenti con la quale potrete facilmente ed in maniera immediata, testare i programmi da voi realizzati.

Come sempre, buon lavoro e buon divertimento con i PIC !

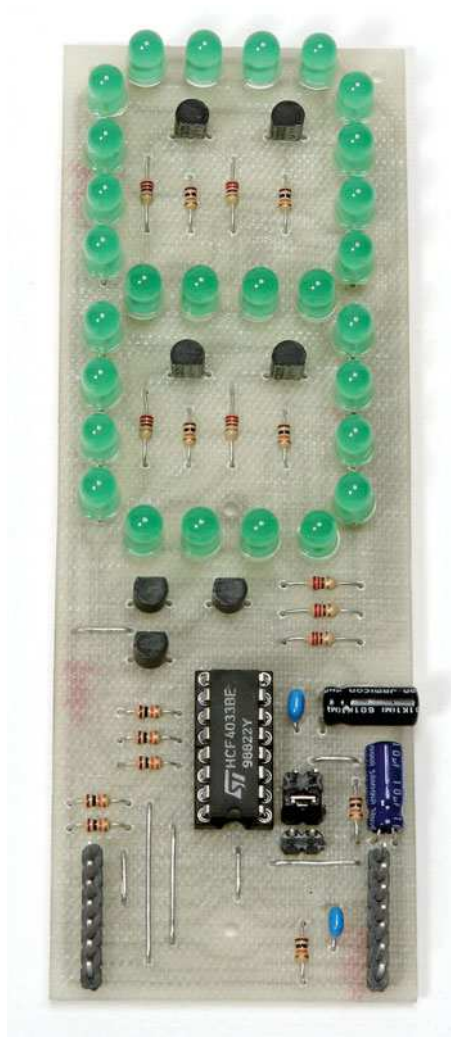
CONTATORE MODULARE A DISPLAY

di Fabrizio Fazio
servizio@pianetaelettronica.it

Ecco un contatore con display a sette segmenti completamente modulare, ossia, con la possibilità di collegare assieme diversi esemplari per realizzare contatori a più cifre. Con questo articolo, oltre a fornire una soluzione circuitale utile in diverse applicazioni, più che descrivere il circuito, molto semplice, vogliamo illustrare il funzionamento dell'integrato 4033.

Il circuito, lo vediamo in queste pagine, è molto semplice: utilizza un integrato dedicato, il CD4033.
Tale integrato, possiede al suo interno tutta la logica che consente di far accendere i segmenti del display formando i numeri, e di far avanzare il numero visualizzato ad ogni impulso applicato al suo ingresso (pin 1).
Per ciascun segmento abbiamo utilizzato 4 diodi led da 5 millimetri. Le dimensioni della cifra sono di circa 4x7 centimetri, tali da consentire normalmente di poter leggere il numero visualizzato fino ad oltre 10 metri di distanza.
I led sono pilotati da transistor, in modo da poter montare, se l'utilizzo del progetto lo rende necessario, dei led ad alta luminosità, aumentando la visibilità del display o delle piccole lampade.

L'INTEGRATO CD4033

Il grafico in figura 2, ci aiuta a comprendere il funzionamento del CD4033, e come utilizzare i terminali di ingresso e uscita di cui dispone.

Premettiamo che il pin CLK è l'ingresso di clock, e quindi è in questo pin che dobbiamo applicare gli impulsi da conteggiare. Nel grafico, ipotizziamo di applicare una serie di impulsi a frequenza fissa. Tali impulsi sono stati numerati da 1 a 25. Abbiamo evidenziato con tre diversi colori i segnali dei diversi terminali dell'integrato:

- In giallo, gli ingressi.
- In verde, le uscite.
- In celeste, le uscite per i segmenti del display.

Normalmente (vedi i primi 12 impulsi), gli ingressi RST, LT e CKI sono posti a livello basso, mentre l'ingresso RBI è posto a livello alto. In tale condizione, il CD4033 conta gli impulsi applicati al pin CLK, e ad ogni fronte di salita di tali impulsi il numero visualizzato sul display s'incrementa di una unità.

A ciascun numero corrisponde la relativa combinazione di segmenti da "a" fino a "g", secondo quanto indicato nel riquadro a destra.

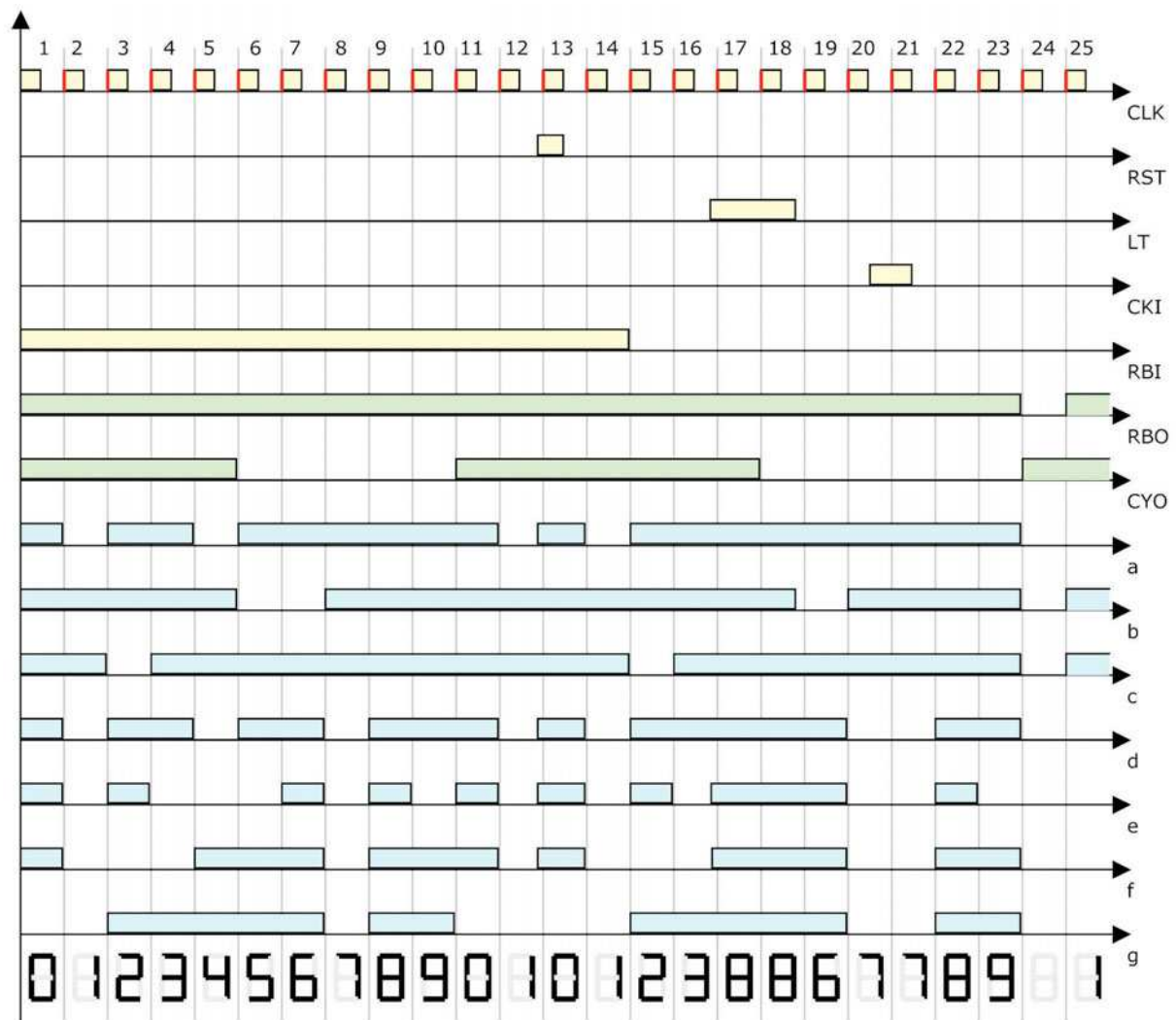
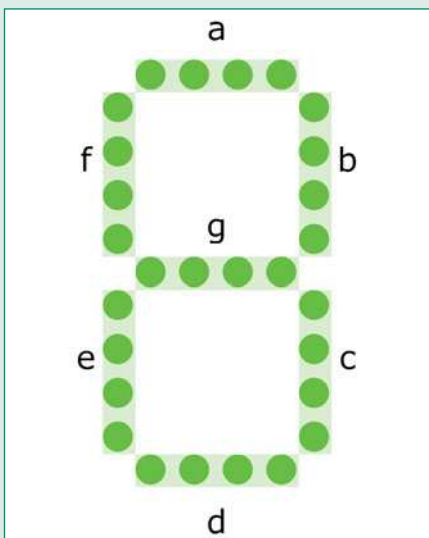



Figura 2: *Funzionamento dell'integrato CD4033*

Il display è formato da sette segmenti, contrassegnati dalle lettere da "a" a "g". Ciascun segmento è formato da 4 diodi led. Accendendo opportunamente i segmenti, è possibile visualizzare tutti i numeri:

- Numero "1" - Segmenti: b, c
- Numero "2" - Segmenti: a, b, g, e, d
- Numero "3" - Segmenti: a, b, c, d, g
- Numero "4" - Segmenti: f, g, b, c
- Numero "5" - Segmenti: a, f, g, c, d
- Numero "6" - Segmenti: g, c, d, e, f, a
- Numero "7" - Segmenti: a, b, c
- Numero "8" - Segmenti: a, b, c, d, e, f, g
- Numero "9" - Segmenti: g, f, a, b, c, d

É bene ricordare sempre che, l'incremento del numero visualizzato, avviene al fronte di salita dell'impulso di clock, ossia, nell'istante in cui il segnale passa da "0" a "1" (istante segnato in rosso nel grafico del segnale su CLK).
Come possiamo vedere, l'ingresso di RST (Reset), se posto a livello "1" azzera il display (vedi impulso n. 13), e finché resta alto il display resta a "0", anche se continuano a giungere gli impulsi su CLK.
Portando a livello alto l'ingresso LT (Lamp Test) si accendono tutti i segmenti del display. Questo pin serve per testare il funzionamento dei led e per tutto il tempo in cui tale pin resta alto, il CD4033 continua a contare, anche se sul display rimane il numero "8". Questo significa che, quando il pin torna basso, il display passa a visualizzare l'effettivo numero a cui è arrivato il conteggio (vedi impulsi n. 17 e 18).

L'ingresso di CKI (clock inhibit) serve invece a bloccare il conteggio. Quindi, se tale pin viene portato a livello alto, il display resta fermo al numero in cui si trova, e il conteggio riprende soltanto quando il pin torna basso.
In questo modo, il contatore ignora tutti gli impulsi di clock che giungono nel periodo in cui il pin CKI è alto (vedi impulso n. 21).
L'uscita CYO (Carry Out) è alta quando sul display è visualizzato un numero compreso tra 0 e 4 ed è bassa quando il numero visualizzato è compreso tra 5 e 9. Dunque su tale pin esce un impulso ogni 10 impulsi di clock e viene utilizzato per pilotare l'ingresso di clock di un secondo CD4033, per realizzare un contatore a 2 cifre. Grazie a questo pin è possibile realizzare un contatore a molte cifre, come nello schema esemplificativo di figura 3, in cui vediamo un contatore a 3 cifre.

Per concludere la descrizione del CD4033, parliamo dei pin RBI ed RBO che, come si vede nel grafico a pagina precedente, funzionano in questo modo:

- Se l'ingresso RBI è posto a livello alto, l'integrato funziona normalmente, e l'uscita RBO è sempre a livello alto.
- Se invece sull'ingresso RBI applichiamo un livello basso, il CD4033 conterà normalmente, ma ogni volta che dovrebbe comparire il numero "0", il display resta spento e l'uscita RBO va a livello basso.

Utilizzando opportunamente i pin RBO ed RBI, possiamo fare in modo che non siano visualizzati gli zeri non significativi. Nel nostro progetto abbiamo impiegato tre connettori strip-line, in uno solo dei quali deve essere inserito un ponticello di corto-circuito.

Come vediamo dallo schema elettrico, con il suddetto ponticello abbiamo la possibilità di scegliere se collegare il pin 3 (RBI) a massa (livello logico 0) oppure al positivo di alimentazione (livello logico 1), oppure al pin 4 del CD4033 relativo alla cifra precedente.

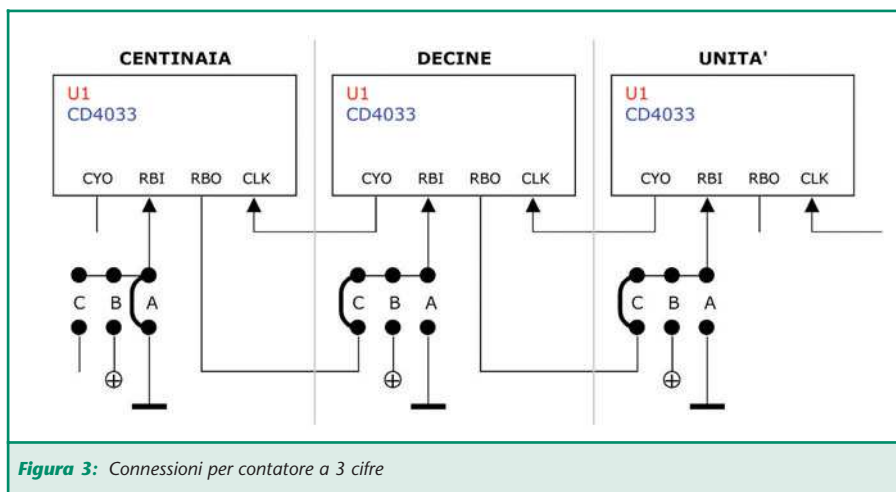
Distinguiamo i tre casi:

1° CASO - Pin 3 a livello alto
In questo caso il contatore funziona normalmente.

2° CASO - Pin 3 a livello basso
In questo caso il display resterà spento ogni volta che dovrebbe visualizzare il numero "0".

3° CASO - Pin 3 al pin 4 della cifra precedente
In questo caso il display resterà spento quando dovrebbe visualizzare il numero "0" soltanto se le cifre precedenti sono tutte "0", a patto che nella scheda della prima cifra, il ponticello sia impostato con "pin 3 a livello basso".
Per meglio comprendere quanto detto, nella tabella che segue vediamo cosa visualizzano i tre display dello schema di figura 3, nel caso in cui ponessimo tutti i ponticelli su "B" (1° caso), nel caso in cui li ponessimo tutti su "A" (2° caso) e infine nel caso in cui li ponessimo esattamente come nello schema in figura 3.

1° CASO (A)	2° CASO (B)	Schema in figura 3
004	4	4
204	2 4	204
200	2	204
035	35	35

Figura 3: *Connessioni per contatore a 3 cifre*

Come vediamo, nel primo caso gli zeri sono sempre visualizzati, nel secondo caso non sono mai visualizzati (tale impostazione è quindi errata: il connettore "A" serve solo per la prima cifra), nell'ultimo caso vengono visualizzati solo gli zeri significativi.

Il reset automatico

Il condensatore C2 serve per azzerare automaticamente il display al momento dell'accensione. Se non ci fosse tale condensatore, all'accensione il display visualizzerebbe un numero casuale. Se non si intende usufruire dell'azzeramento automatico, C2 può tranquillamente non essere montato. Tale condensatore non deve essere montato se si collega a questo progetto una scheda esterna che provvede già a fornire gli impulsi di reset, ad esempio se colleghiamo l'applausometro che abbiamo presentato il mese scorso.

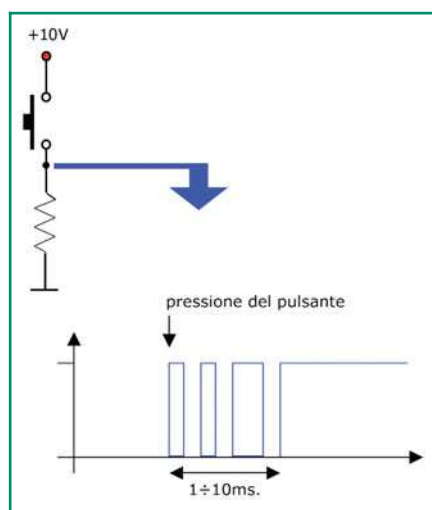



Figura 4: *Il rimbalzo di un pulsante*

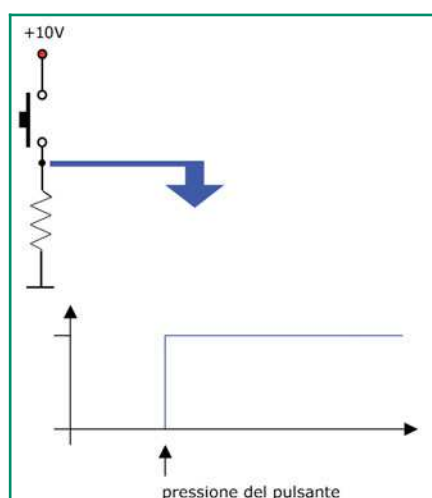



Figura 5: *La funzione antirimbalzo*

Antirimbalzo

Per concludere la descrizione dello schema elettrico, vorremmo spendere due parole sul condensatore C4, che svolge la funzione di antirimbalzo, e che deve essere montato sulla scheda soltanto se si utilizza un pulsante per l'avanzamento del contatore.
Premettiamo che tutti i pulsanti, quando vengono premuti, generano il cosiddetto rimbalzo, che consiste nel fatto che il contatto non si chiude istantaneamente in modo netto, ma per un breve periodo è come se il pulsante si aprisse e chiudesse più volte, generando cioè degli impulsi spuri come vediamo in figura 4.
Se accendiamo una lampadina con un pulsante non ce ne accorgiamo, perchè il rimbalzo dura normalmente pochi millesimi di secondo (cioè pochi millisecondi), ma il nostro CD4033 se ne accorge e li conta.
Per esperimento, provate a collegare un pulsante sulla nostra scheda senza montare C4, e noterete che ad ogni pressione il display anziché avanzare di una sola cifra, va avanti di più cifre.
Con il condensatore C4 questo non succede (figura 5), perchè tale condensatore simula il rilascio ritardato del pulsante. Sempre in merito all'esperimento, è bene utilizzare un pulsante e non un semplice filo da collegare "a mano", perchè quest'ultimo genera impulsi spuri con tempi diversi, rispetto al pulsante.
É possibile realizzare circuiti antirimbalzo più complessi, ma quello da noi adottato è il più semplice, e comunque efficace.
É chiaro che C4 non consente di conteggiare pressioni troppo rapide del pulsante. Utilizzando il valore da noi proposto di 10 µF il CD4033 accetta fino a circa 3-4 pressioni al secondo. Diminuendo la capacità di C4, l'integrato accetta un maggior numero di pressioni al secondo, ma l'effetto dell'antirimbalzo peggiora, e viceversa aumentando la capacità.

REALIZZAZIONE PRATICA

Premettiamo che se vogliamo montare più schede, per realizzare un contatore a più cifre, i componenti R2, R3, R4, C2 dovranno essere montati solo su una qualsiasi delle varie schede.
La tensione di alimentazione deve essere compresa tra 9 e 15 V. I valori delle resistenze da R12 a R18 sono calcolati per una tensione di 12 V. In caso di alimentazione a 9V, le suddette resistenze dovranno essere ridotte a 68 Ω, per 10 V a 120 Ω e per 15 V devono essere portate a 470 Ω da 1/4 di Watt.

É possibile realizzare un display ancora più grande di quello proposto in queste pagine, sfruttando la stessa scheda (ovviamente i led più grandi o le lampade dovranno essere montate su un'altra scheda più grande, oppure su un supporto meccanico). Ad esempio, si può realizzare un display da 75x140 mm., visibile a oltre 20 metri di distanza, sfruttando lo stesso schema, ma utilizzando dei led da 10 millimetri di diametro.
Si possono impiegare anche piccole lampade; in ogni caso occorre considerare due cose:

La tensione

Tra il positivo di alimentazione e il collettore di ciascun transistor, (dove cioè va collegato il ramo di led o lampade), è presente la stessa tensione applicata in alimentazione, quando il relativo segmento è acceso. Quindi la tensione delle lampade

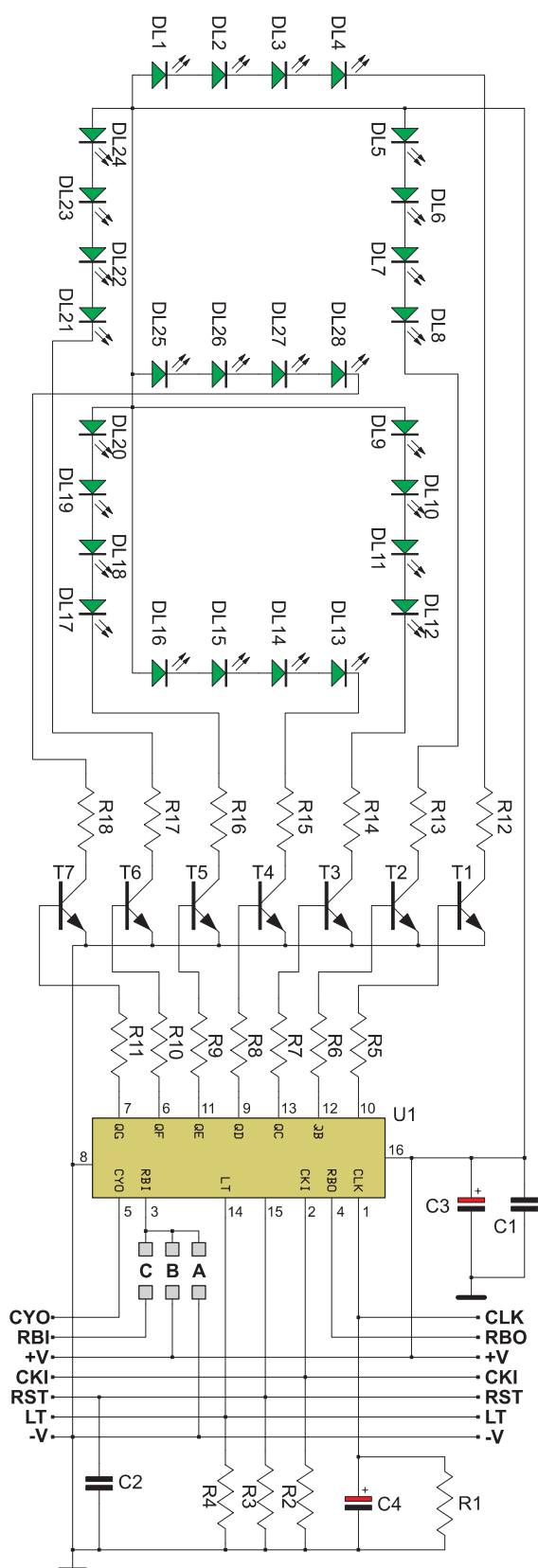



Figura 6: Schema elettrico

e quella di alimentazione devono essere scelte opportunamente.

La corrente

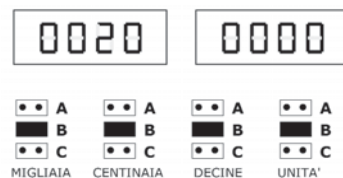
Ciascun segmento può accettare un carico di:

- Max. 80 mA se l'alimentazione è da 9 V
- Max. 100 mA se l'alimentazione è da 12 V
- Max. 130 mA se l'alimentazione è da 15 V

Per il montaggio, consigliamo come sempre di iniziare dai componenti a basso profilo, prestando attenzione al corretto inserimento dei componenti polarizzati.

Per realizzare un contatore a più cifre, le schede vanno collegate semplicemente come illustrato nello schema pratico di figura 9, tramite i connettori a 7 poli.

Impostazione dei ponticelli

Per visualizzare sempre gli zeri:
Ad esempio, ecco come impostare i ponticelli per visualizzare come segue i numeri "20" e "0", con 4 cifre:

Per visualizzare solo gli zeri significativi:
Ad esempio, ecco come impostare i ponticelli per visualizzare come segue i numeri "20" e "0", con 4 cifre:

Attenzione: in questo caso, quando il valore è "0", tutte le cifre sono spente. Se si vuole che l'ultimo display visualizzi lo zero, il ponticello della relativa scheda

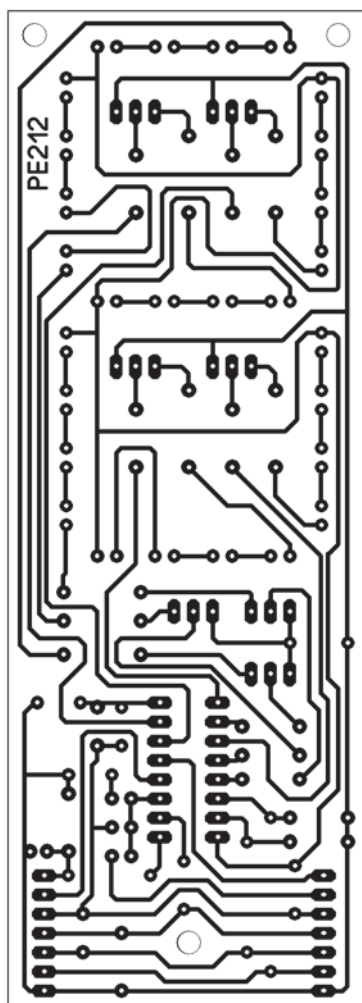



Figura 7: Circuito stampato scala a:a (lato rame)

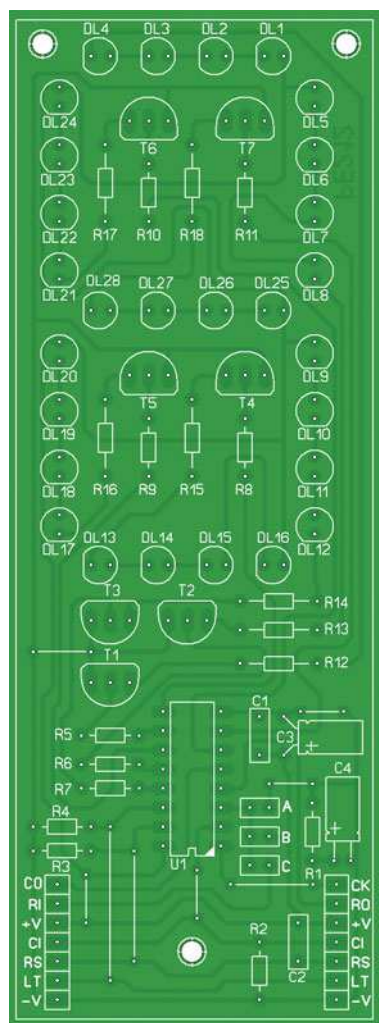



Figura 8: Posizionamento dei componenti

Elenco componenti	
Sigla	**Valore**
R1÷R11	10 KΩ 1/8 W
R12÷R18	270 Ω 1/8 W
C1, C2	100 nF multistrato
C3	100 μF 25 V elettrolitico
C4	10 μF 25 V elettrolitico
DL1÷DL28	Diodo led verde 5mm.
T1÷T7	Transistor BC337
U1	Integrato CD4033
Varie	
Zoccolo DIP 16 pin	
2 connettori strip-line a 7 poli	
3 connettore strip-line a 2 poli	
Ponticello di cortocircuito	

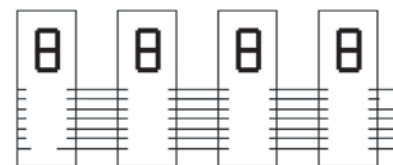



Figura 9: Realizzazione di un contatore a più cifre

deve essere posto su "B", come di seguito indicato:

Per contare fino a 10, 100, 1000...
In alcune applicazioni, ad esempio se utilizziamo questo progetto in unione all'applausometro presentato sul numero di Febbraio (224) si può avere la necessità di un fondo scala di 10, 100, 1000, ecc. anziché 9, 99, 999, ecc. In questo caso si dovrà realizzare una ulteriore scheda PE0212, in cui andranno montati soltanto gli 8 led relativi ai segmenti "b" e "c". Anche i transistor e le resistenze relative agli altri segmenti non andranno montate.
Per fare in modo che il conteggio si blocchi a "100", occorre però una semplice modifica: inserire un diodo 1N4150 con l'anodo sul pin 10 e il catodo sul pin 2 di U1. Tale diodo può essere saldato direttamente sul lato piste della scheda (ovviamente soltanto sulla scheda della prima cifra in cui sono montati gli 8 led).

Utile a sapersi...
É possibile utilizzare dei led ad alta luminosità, tenendo presente che hanno un angolo di visualizzazione piuttosto ristretto (sono molto luminosi, ma solo se visti di fronte, quindi adatti per essere visti a grandi distanze). A chi vuole utilizzare i classici led economici, ricordiamo che il colore più visibile a distanza è normalmente il **verde**, e il meno visibile è il rosso.
Consigliamo di acquistare i 28 led tutti della stessa marca, e possibilmente dello stesso lotto, perchè altrimenti si potrebbero avere alcuni led più luminosi ed altri meno. Anche led dello stesso lotto possono avere piccole differenze, ma sono normalmente molto più contenute e accettabili.

In figura 13 riportiamo le connessioni esterne della scheda. Al pin 1 (ingresso di clock) devono essere applicati gli impulsi da conteggiare. La tensione di alimentazione deve essere applicata tra i pin 3 (+) e 7 (-).
Gli ingressi CKI, RST e LT (pin 4, 5 e 6) sono disponibili per essere pilotati esternamente. Le loro funzioni sono state illustrate nel capitolo "Il circuito". Per un buon funzionamento, le tensioni da applicare a tali ingressi per livello logico "1" devono essere superiori al 70% della tensione di alimentazione. Il pin 2 non deve essere utilizzato esternamente (serve solo per il collegamento tra le varie cifre).

In figura 14 vediamo come collegare il contatore all'applausometro che abbiamo presentato il mese scorso, in queste pagine. In questo caso, non dimentichiamo la piccola modifica descritta poco fa per far bloccare il conteggio a 100.

Per chi vuole impiegare questo progetto come contatore eliminacode per esercizi pubblici, riportiamo lo schema in figura 15, dove per brevità abbiamo riportato solo una cifra, ma anche qui abbiamo la possibilità di avere più cifre.

Buon lavoro.

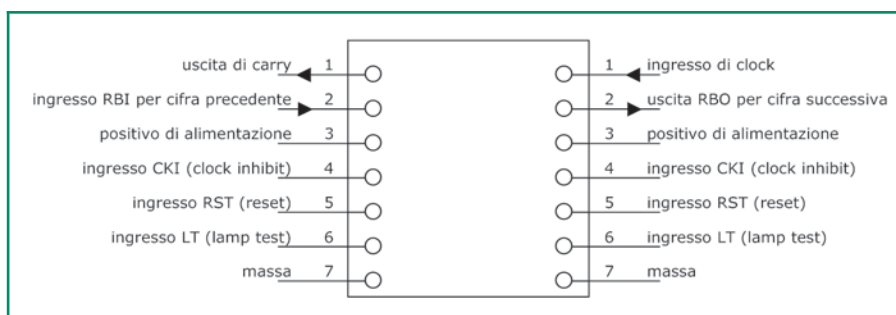



Figura 13: Connettori della scheda

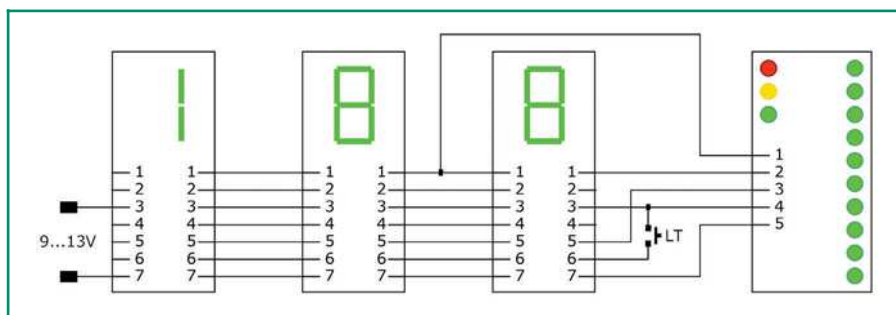



Figura 14: Connessione dell'applausometro

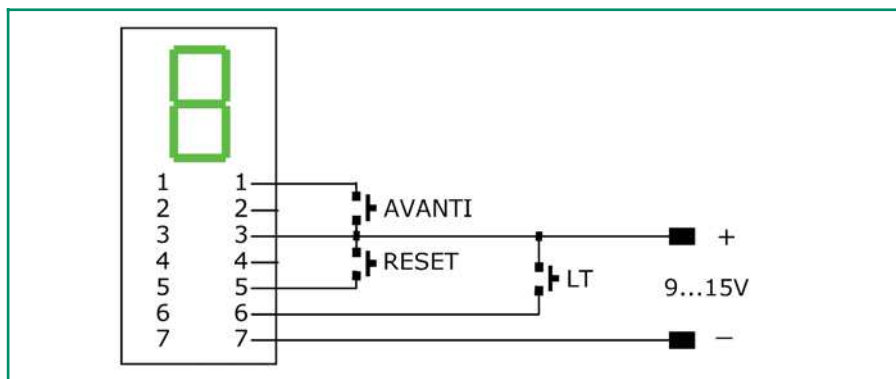



Figura 15: Contatore eliminacode per esercizi pubblici

SILVI MARINA (TE) - FIERA ADRIATICA
S.S.16 (Nazionale Adriatica) - Km. 432

1ª FIERA MERCATO DELL'ELETTRONICA

RADIOAMATORE
COMPUTER
INTERNET
TELEFONIA
ANTENNE
TV SAT
EDITORIA

17 - 18 APRILE 2004

SABATO 9:15 - 19:00 / DOMENICA 9:00 - 19:00
AMPIO PARCHEGGIO e SERVIZIO NAVETTA GRATUITI
RISTORANTE - SELF SERVICE INTERNO

con il patrocinio di

Provincia di Pescara

Comune di Silvi

Sezioni ARI di Chieti e Pescara

Studio Organizzatore Via Siena, 22 - 65122 PESCARA
Tel. 085 4215840 - Fax 085 290358
e-mail: e.pescarafiere@libero.it

ideazione: I6DQD - realizzazione grafica: IZ6BMP

SPICE: LA SIMULAZIONE DEI CIRCUITI ELETTRICI ED ELETTRONICI

terza parte

di Nico Grilloni
nicogrilloni@tin.it

Analizzata la simulazione con Spice di alcuni circuiti in corrente continua e in corrente alternata si espongono qui nuove applicazioni in grado di porre in evidenza le ulteriori potenzialità e l'enorme flessibilità di impiego del software.

Le prime versioni di Spice elaboravano fondamentalmente, nell'analisi di un circuito, le risposte in ampiezza, frequenza e fase e avevano una library di componentistica piuttosto limitata. Nel volgere di un decennio il software si è notevolmente espanso consentendo oltre allo sweep sui parametri elettrici dei circuiti e sulla temperatura di funzionamento, la simulazione Monte Carlo, l'analisi di Fourier, l'analisi del rumore e della distorsione. Questa espansione del processo di simulazione ha decretato il successo di Spice che, come si è già anticipato, al presente viene considerato come strumento insostituibile nell'analisi di un qualsiasi circuito sia analogico che digitale, che analogico-digitale.

LA SIMULAZIONE MONTE CARLO

È una delle simulazioni più interessanti e anche fra le più indicate per la verifica del funzionamento reale di un circuito dal momento che consente di tener conto della tolleranza dei componenti inseriti.

Si consideri come primo esempio il filtro attivo di cui alla figura 1. Si tratta di un filtro passabanda di tipo MFB, ossia a reazione multipla, con attenuazione di 20 dB/decade, con frequenza f_0 di centrobanda di 10 kHz, guadagno in banda passante pari a 5 e fattore di merito *Q* pari a 10.

La risposta in frequenza del filtro è riportata nella figura 2. Qui, in fase di simulazione si è evidenziata la banda passante a –3 dB tramite la reciproca posizione dei marker verticali *a* e *b*. Si legge: $(a - b) = 1{,}004$ kHz. Poiché il fattore di merito è il rapporto fra la frequenza di centrobanda f_0 = 10 kHz e la banda passante *B*, per *Q* si ha il valore: $Q = 10000/1004 = 9{,}96 \cong 10$. La posizione del marker orizzontale *d* fornisce subito il guadagno alla frequenza f_0. Si legge: *Yd* = 14,5 dB. A 14,5 dB corrisponde un guadagno puramente numerico pari a 5,3. La risposta in frequenza è, pertanto, perfettamente aderente ai dati di progetto del filtro.

Attivando la simulazione Monte Carlo si è lasciata all'1% la tolleranza delle resistenze, mentre alle capacità *C1* e *C2* si è attribuita una tolleranza del 25% che rappresenta un valore abbastanza consueto a meno di non usare condensatori di alta e altissima qualità. Si è così ricavato il diagramma di cui alla figura 3 dove i marker verticali *a* e *b* indicano, rispettivamente, una frequenza di centrobanda *Xa* = 10,68 kHz e *Xb* = 9,06 kHz, ossia valori che si discostano notevolmente dal dato fondamentale di progetto che impone una fo = 10 kHz.

Un valore accettabile della frequenza di centrobanda, cioè non eccessivamente discosto da 10 kHz, si è ottenuto attribuendo alle capacità una tolleranza del 10%.

Nello stesso diagramma si vede che anche il guadagno subisce una

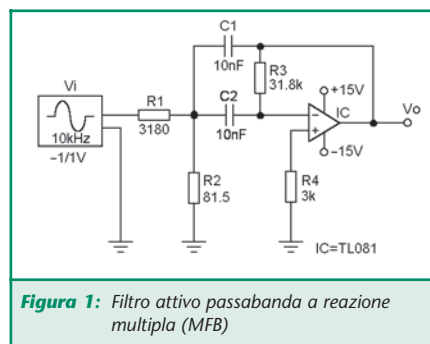



Figura 1: *Filtro attivo passabanda a reazione multipla (MFB)*

diminuzione nello scostamento dalla frequenza per la quale il filtro è stato dimensionato, mentre la banda passante *B*, essendo fornita dall'espressione:

$$B = 0{,}318 / (R_3 \cdot C_1)$$

si contrae per C_1 = 0,033 μF (+25%) e si dilata per C_1 = 0,033 μF (–25%). La figura 4 mostra quindi un semplice circuito *RC* nel quale si è attivata la funzione Monte Carlo di Spice nell'analisi del transitorio di carica. L'indicatore (.IC), posto fra la resistenza e la capacità, impone la condizione iniziale V_c = 0, ossia che alla chiusura del deviatore *S* il condensatore sia scarico. Nel menu relativo alla simulazione in oggetto si è lasciata invariata all'1% la tolleranza della resistenza R_1 e si è attribuita una tolleranza del 25% alla capacità del condensatore C_1. Ciò significa che durante la simulazione R_1 mantiene pressoché costante il suo valore di 100 kΩ, mentre il valore della capacità varia in modo del tutto casuale nell'intorno della tolleranza imposta attivando il numero di simulazioni definito nel menu relativo.
Le curve fornite da Spice sono riportate nel diagramma di cui alla figura 5.
Le curve (a) e (b) si riferiscono a valori di C_1 rispettivamente eguali a 7,5 μF e 12,5 μF. ossia ai valori estremi che assume la capacità C_1 per una tolleranza del 25%. L'esattezza di queste curve può essere verificata ricordando che l'espressione che fornisce il transitorio della tensione V_c di carica di un condensatore è:

$$V_c = V_i \cdot \left(1 - e^{-\frac{t}{\tau}}\right)$$

Considerando, per esempio, l'istante *t* = 1 s indicato dal marker verticale *a* e tenendo conto che la costante di tempo τ, per R_1 = 100 kΩ e C_1 = 7,5 μF,

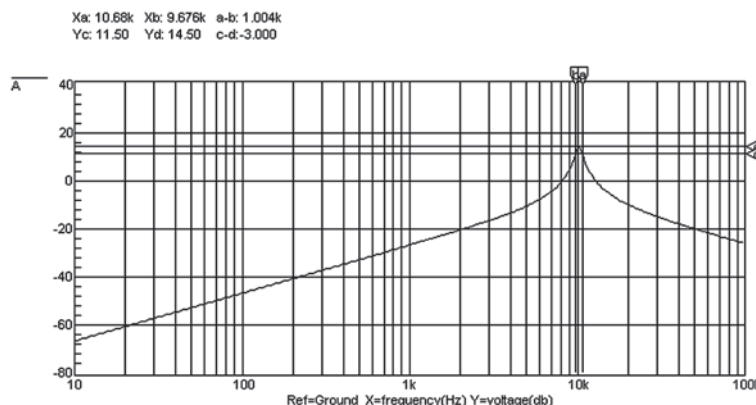



Figura 2: *Risposta del filtro passabanda di cui alla figura 1. La frequenza f_0 di centrobanda è di 10 kHz. La distanza (a – b) fra i marker verticali indica la banda passante B pari a circa 1 kHz. Il filtro è stato, infatti, dimensionato per un fattore di merito eguale a 10*

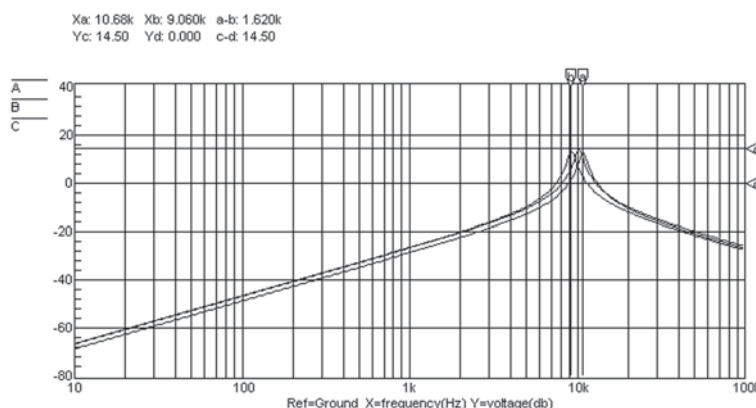



Figura 3: *Attivando la simulazione Monte Carlo di Spice attribuendo alle due capacità del filtro attivo di cui alla figura 1 una tolleranza del 25 %, nella risposta in frequenza si nota un notevole scostamento della frequenza f_0 dal valore per il quale il filtro è stato dimensionato*

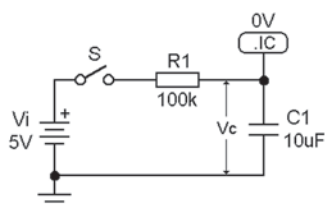



Figura 4: *Circuito RC al quale si è applicata la simulazione Monte Carlo per valutare le curve del transitorio di carica al variare del valore nominale della capacità C a cui è stata attribuita la tolleranza del 25 %*

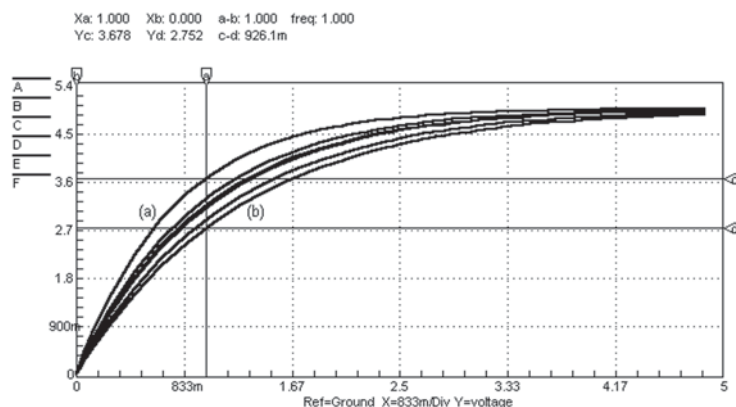



Figura 5: *Curve del transitorio di carica per il circuito RC di cui alla figura 4 al variare del valore nominale della capacità*

è $\tau = R_1 \cdot C_1 = 100000 \times 7{,}5 \times 10^{-6} =$ 750 ms, mentre per $R_1 = 100$ kΩ e $C_1 = 12{,}5$ µF è $\tau = R_1 \cdot C_1 = 100000$ _ $12{,}5$ _ $10^{-6} = 1{,}25$ s, per $V_i = 5$ V, applicando l'espressione precedente ai due casi limite si ricava rispettivamente:

$V_{c1} = 3{,}68$ V $\qquad V_{c2} = 2{,}753$ V

Questi due dati sono indicati nel diagramma di cui alla figura 5 dai marker orizzontali *c* e *d*. Si legge infatti *Yc* = 3,678 V e *Yd* = 2,752 V.

SPICE NEI CIRCUITI RADDRIZZATORI

Si consideri il circuito di cui alla figura 6 rappresentativo di un raddrizzatore a una semionda.
Con il deviatore *S* aperto Spice consente di ricavare le forme d'onda di cui alla figura 7 dove in grassetto è riportata la tensione V_o di uscita, ossia la tensione ai capi del carico R_L, mentre è a tratto normale la tensione V_i applicata in ingresso.
Quest'ultima ha valore di 20 V da picco a picco (10 V di valore massimo) e frequenza di 50 Hz. La tensione V_o di uscita, che ha ovviamente la medesima frequenza, per l'azione raddrizzatrice e per la disposizione del diodo *D* si evolve esclusivamente nel semipiano positivo e presenta un valore massimo appena inferiore al valore massimo della V_i. Come, infatti, indica la reciproca posizione dei marker *c* e *d* - si legge (*c* – *d*) = 660,7 mV - la V_o è, appunto, inferiore alla V_i di circa 0,7 V. Ciò è dovuto alla caduta di tensione sul diodo che può variare fra 0,6 V e 0,7 V.
Chiudendo il deviatore *S* si pone in parallelo al carico la capacità *C* = 10 µF. La tensione V_o di uscita fornita da Spice ha allora l'andamento illustrato nella figura 8 dove è evidente l'azione livellatrice del condensatore sebbene di non elevata capacità.

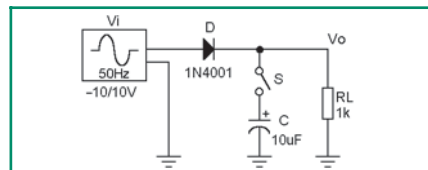



Figura 6: *Circuito raddrizzatore a una semionda*

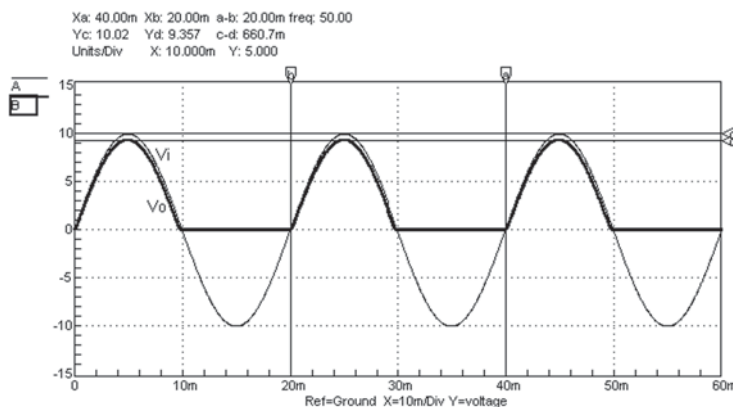



Figura 7: *Andamento della tensione di ingresso V_i e della tensione di uscita V_o ai capi del carico R_L nel circuito raddrizzatore di cui alla figura 6 tenendo S aperto*

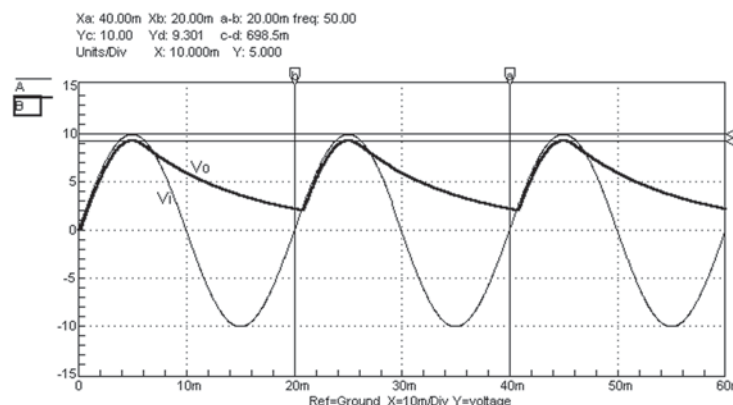



Figura 8: *Andamento della tensione di ingresso V_i e della tensione di uscita V_o ai capi del carico R_L nel circuito raddrizzatore di cui alla figura 6 con S chiuso*

Figura 9: *Circuito raddrizzatore a onda intera*

Figura 10: *Andamento della tensione V_o ai capi del carico R_L nel circuito raddrizzatore di cui alla figura 9 tenendo S aperto*

La figura 9 riporta quindi un circuito analogo, ma nel quale la presenza del ponte di Graetz realizza il raddrizzamento a onda intera (anche detto a due semionde). Con *S* aperto la tensione V_o di uscita ha l'andamento di cui alla figura 10. Con *S* chiuso, e quindi con la capacità da 50 μF in parallelo al carico, per la V_o si ha l'andamento riportato in grassetto nella figura 11. In entrambi i diagrammi delle figure 10 e 11 si vede che il valore massimo della tensione di uscita è di circa 8,6 V mentre il valore massimo della V_i di ingresso è 10 V. La differenza 10 – 8,6 = 1,4 V indica la caduta di tensione nei diodi del ponte che, in un periodo del segnale di ingresso, conducono in due per volta e quindi la caduta di tensione dell'uno va a sommarsi alla caduta di tensione dell'altro (0,7 x 2 = 1,4 V).

Se adesso, tramite Spice, si esegue un sweep attribuendo alla capacità valori variabili, per esempio da 50 μF a 250 μF per gradini di 100 μF, si ha una diminuzione del fattore di ripple *F*, ossia dell'ondulazione della V_o, via via che la capacità assume valore più elevati. Questo risultato è espresso da Spice tramite le curve riportate nella figura 12 dove, dal basso verso l'alto, sono rappresentate la forma d'onda in uscita in assenza della capacità, e quindi le forme d'onda della stessa V_o per valori di *C* pari, rispettivamente, a 50 μF, 100 μF e 250 μF. È evidente come all'incremento della capacità corrisponda una diminuzione del ripple. Questo risultato era ovviamente prevedibile sia sperimentalmente che analiticamente. L'espressione del fattore *F* di ripple, per in raddrizzamento a onda intera, è infatti:

$F = 2{,}9 / (C \cdot R_L)$

Pertanto, a parità di carico R_L, essendo *F* inversamente proporzionale alla capacità *C*, era del tutto intuitivo che all'aumentare di *C* dovesse diminuire *F*.

L'ANALISI DI FOURIER

È noto che l'analisi armonica di un segnale periodico non può prescindere dallo sviluppo in serie di Fourier secondo il quale una funzione periodica, per esempio della variabile indipendente *t*, può essere considerata come somma di diverse componenti sinusoidali di cui quella definita fondamentale ha la stessa

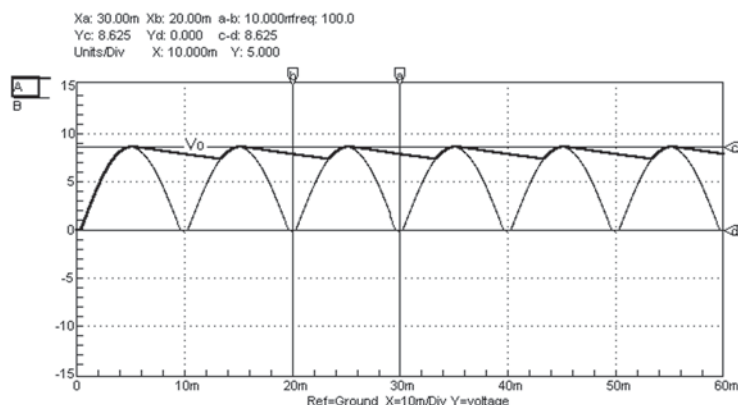



Figura 11: *In grassetto l'andamento della tensione ai capi del carico R_L nel circuito raddrizzatore di cui alla figura 9 con S chiuso*

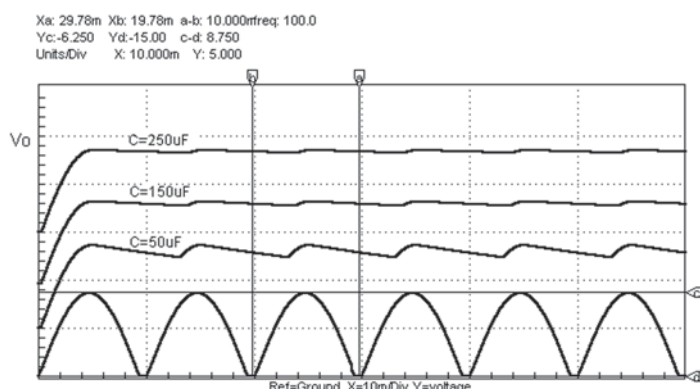



Figura 12: *Attribuendo alla capacità di filtro del circuito di cui alla figura 10 valori variabili fra 50 μF e 250 μF, tramite la funzione sweep di Spice si ottengono i diagrammi di figura. Si noti come all'aumentare del valore di C si abbia una sempre più marcata riduzione del ripple*

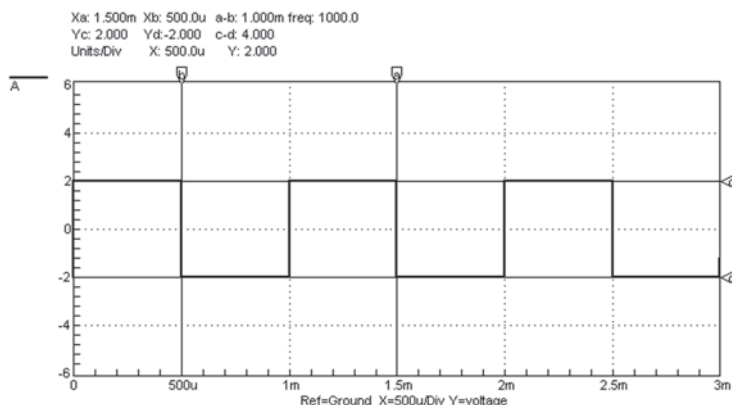



Figura 13: *Segnale a onda quadra alternato di frequenza f = 1 kHz indicata dalla reciproca posizione dei marker verticali a e b*

frequenza f del segnale originario, mentre le successive, definite armoniche, hanno frequenza doppia, tripla, quadrupla, ecc. Il segnale periodico può quindi essere studiato come somma di un termine costante e di una serie infinita di termini sinusoidali la cui frequenza è multipla intera della frequenza del segnale.

Non essendo questa la sede per approfondire l'argomento, diciamo soltanto che per un segnale alternato a onda quadra di frequenza pari, per esempio, a 1 kHz e la cui ampiezza - figura 13 - possa, per esempio, variare fra –2 V e +2 V, il valor medio sarà nullo e saranno presenti solo le armoniche di ordine dispari.

Spice, una volta impostata la relativa analisi, evidenzia questo risultato come mostra la figura 14 dove, per l'appunto, sono presenti solo le armoniche di frequenza 1 kHz, 3 kHz, 5 kHz, ecc., mentre sono assenti le armoniche di ordine pari. Nello stesso diagramma si vede altresì come l'ampiezza delle varie armoniche decresca all'aumentare della frequenza. Quest'ultima considerazione consente di ritenere nulle le armoniche del segnale da un certo termine in poi e di limitare quindi l'analisi ad un prestabilito numero di armoniche.

Se adesso si considera il segnale a onda quadra di cui alla figura 15, si vede che questo è periodico di ampiezza V_i, frequenza $f = 1$ kHz, ma non alternato. In tal caso, lo sviluppo in serie di Fourier - figura 16 - indica, oltre alla presenza delle sole armoniche di ordine dispari, anche la presenza di un valor medio diverso da zero di frequenza $f = 0$ e di ampiezza pari a $V_i/2$. Questo termine è posto in evidenza dal marker verticale *b*.

Si considerino quindi i due circuiti esposti nella figura 17. Il circuito della figura 17 *a* è un classico filtro passa basso *RC*, mentre il circuito della figura 17 *b* è un classico filtro passa alto. La sollecitazione è il segnale V_i a onda quadra di ampiezza 0 V ÷ 5 V e frequenza $f = 1$ MHz. Eseguendo l'analisi di Fourier si ricava lo spettro di frequenze di cui alla figura 18 dove dall'alto verso il basso sono riportati lo spettro della V_i e quindi della tensione V_c ai capi del condensatore *C* e della tensione V_r ai capi della resistenza *R*.

Si può facilmente constatare come,

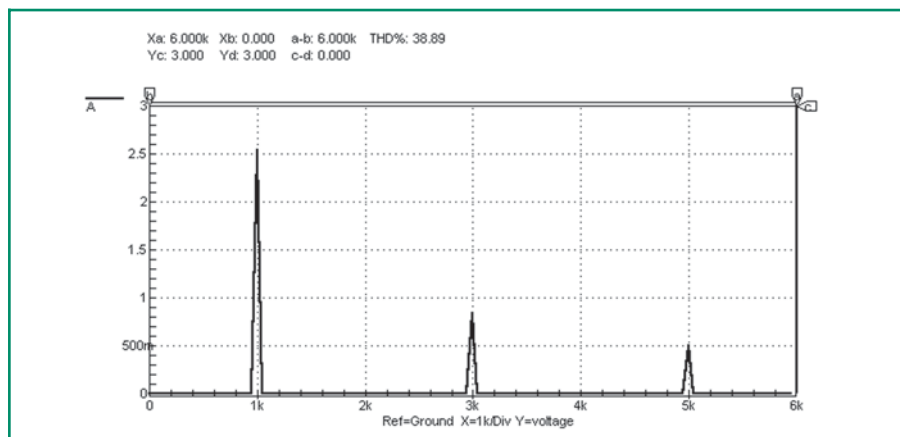



Figura 14: *Spettro di frequenza per la forma d'onda di cui alla figura 13 ricavato attivando l'analisi di Fourier di Spice*

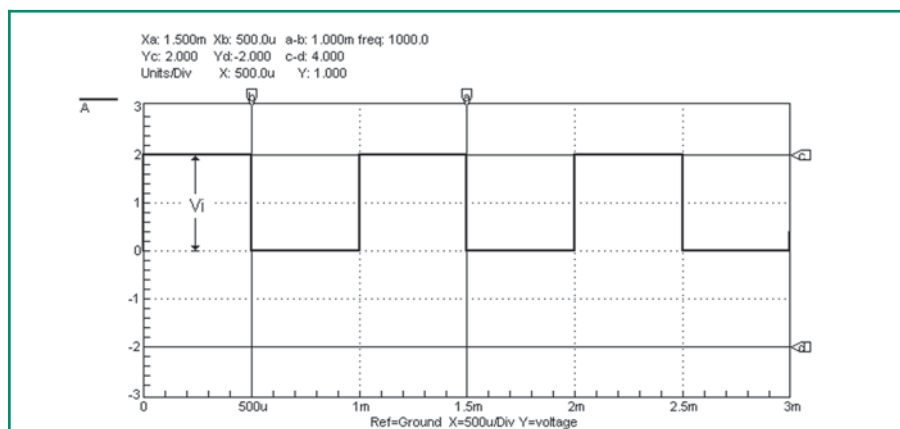



Figura 15: *Segnale a onda quadra con ampiezza (0 ÷ 2)V e frequenza f = 1 kHz*

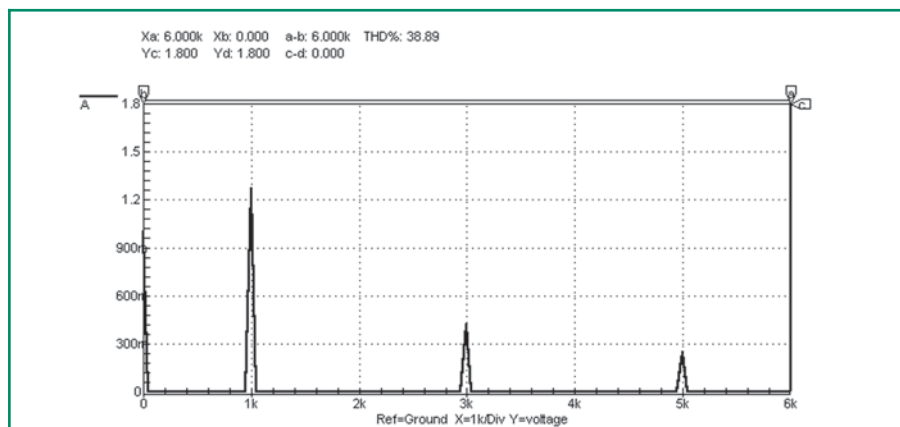



Figura 16: *Spettro di frequenza per la forma d'onda di cui alla figura 15 ricavato attivando l'analisi di Fourier di Spice*

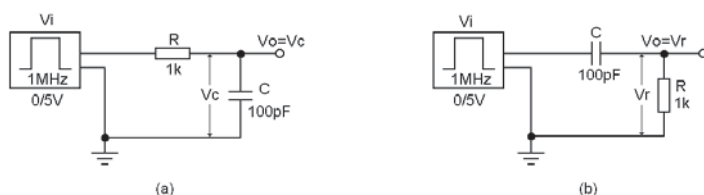



Figura 17: *In (a) un filtro passa-basso e in (b) un filtro passa-alto. Entrambi i filtri presentano una frequenza di taglio pari a 1,59 MHz*

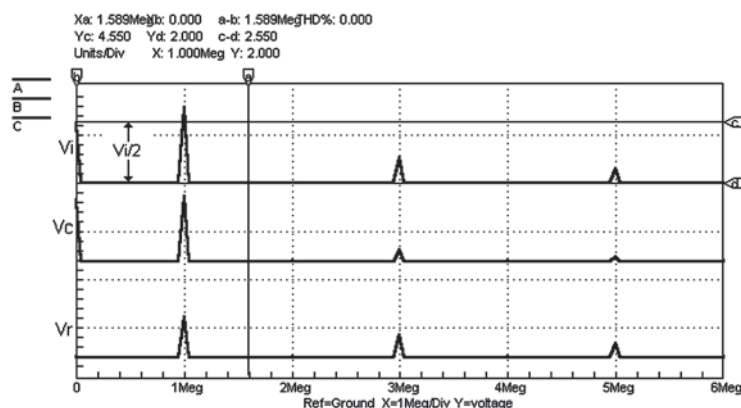



Figura 18: *Spettri di frequenza relativi a V_i, V_c e V_r con riferimento ai circuiti di cui alla figura 17*

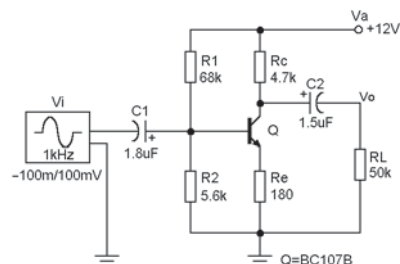
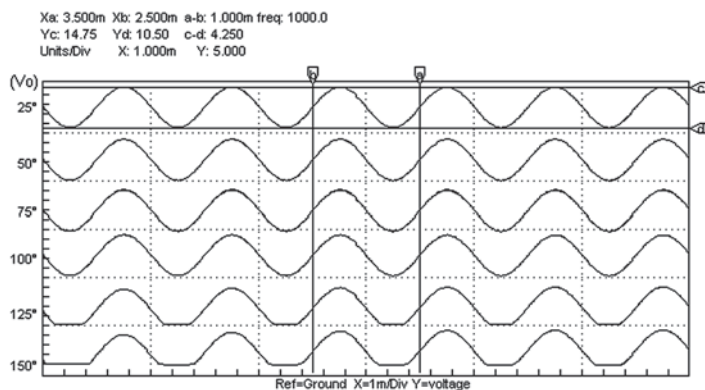



Figura 19: *Stadio amplificatore a BJT*

aldilà della frequenza di taglio pari a 1,59 MHz e indicata dal marker verticale *a*, prelevando l'uscita ai capi del condensatore - figura 17 *a* - l'ampiezza delle armoniche subisca un rapido decremento: all'aumentare della frequenza, infatti, il comportamento della capacità approssima la condizione di cortocircuito sull'uscita. Si noti altresì la presenza della componente di valor medio a frequenza zero indicata dal marker verticale *b* che in questo caso è presente dal momento che il segnale V_i è periodico ma non alternato.
Viceversa, prelevando l'uscita ai capi della resistenza - figura 17 *b* - poiché la reattanza capacitiva diminuisce all'aumentare della frequenza, le armoniche del segnale di uscita V_r presentano un'ampiezza sempre più prossima all'ampiezza delle armoniche relative al segnale di ingresso V_i. Nel diagramma relativo alla V_r si nota altresì come sia assente la componente continua che viene, infatti, "tagliata" dalla capacità.

TEMPERATURE SWEEP

Per verificare il funzionamento dello sweep in temperatura si è fatto ricorso al circuito amplificatore a BJT di cui alla figura 19. Il guadagno di tensione *G* di questo stadio è, con buona approssimazione, pari al rapporto fra la resistenza di collettore R_c e la resistenza di emettitore R_e ed è quindi assai prossimo a 26. In fase di simulazione con Spice si è imposta una variazione della temperatura di funzionamento del circuito da 25 °C a 150 °C per gradini di 25 °C. Il diagramma che si è ricavato per la tensione di uscita V_o è riportato nella figura 20 dove si può constatare che la forma d'onda non presenta distorsioni notevoli per temperature comprese fra 25 °C e 50 °C, mentre già a 75 °C si presenta una lieve tosatura della semionda negativa, tosatura che poi diviene molto evidente per temperature comprese fra 100 °C e 150 °C.
Sempre utilizzando il circuito di cui alla figura 19, si è poi imposta una temperatura costante di funzionamento di 27 °C imponendo altresì una variabilità della resistenza R_e da 80 Ω a 200 Ω per gradini di 40 Ω. La variabilità della resistenza di emettitore implica, per quanto su si è detto, la variabilità del guadagno (che è inversamente proporzionale alla R_e).
Il diagramma che si è ricavato per la

Figura 20: *Andamento della tensione V_o ai capi del carico R_L dello stadio di cui alla figura 19 dopo aver imposto una variazione di temperatura fra 25 °C e 150 °C per gradini di 25 °C*

tensione V_o di uscita è esposto nella figura 21. Qui si vede che per valori della resistenza di emettitore di 80 Ω e 120 Ω il segnale, a causa dell'elevata amplificazione, presenta una notevole distorsione (tosatura della semionda negativa), mentre già per R_e = 160 Ω il segnale appare accettabile e diviene ottimo per R_e = 200 Ω. La figura 22 riporta, quindi, la risposta in ampiezza dello stadio di cui alla figura 19 per R_e = 180 Ω. Qui la distanza fra i marker orizzontali *c* e *d* indica un valore da picco a picco della V_o pari a 4,747 V. Poiché lo stadio è stato pilotato con una V_i = 200 mV = 0,2 V da picco a picco, il guadagno $G = V_o / V_i$, ha il valore:

$$G = 4{,}747 / 0{,}2 = 23{,}7$$

che, espresso in dB, diviene:

$$G_{dB} = 20 \log 23{,}7 = 27{,}49 \text{ dB}$$

Figura 21: *Andamento della tensione V_o ai capi del carico R_L dello stadio di cui alla figura 19 dopo aver imposto una variazione della resistenza di emettitore R_e da 80 Ω a 200 Ω.*

Figura 22: *Risposta in ampiezza, fornita da Spice, dello stadio di cui alla figura 19 attribuendo alla resistenza di emettitore il valore di 180 Ω*

Figura 23: *Risposta in frequenza, fornita da Spice, dello stadio amplificatore a BJT di cui alla figura 19*

Questo risultato è posto in evidenza dal diagramma della risposta in frequenza ricavato da Spice e riportato nella figura 23. La posizione del marker orizzontale *d* esprime, infatti, un guadagno in banda passante pari a 27,5 dB (si legge, infatti, *Yd* = 27,5 dB). La frequenza di taglio inferiore è fornita, a –3 dB, dalla posizione del marker verticale *a* che indica una frequenza pari a 18,53 Hz.

Con questa terza parte si conclude la serie degli articoli dedicati all'illustrazione della simulazione al computer di alcuni circuiti elettrici ed elettronici tramite il software Spice.
Speranza dell'autore è di aver sollecitato la curiosità del Lettore per questo metodo di studio, di analisi e di verifica del funzionamento dei circuiti. Coloro che eseguiranno il download del programma dal sito **microcode.com** potranno ripetere le simulazioni qui eseguite come potranno, ancora più proficuamente, realizzare diverse configurazioni circuitali e nuove simulazioni.
L'autore sarà sempre a disposizione del Lettore per ulteriori spiegazioni volte ad approfondire sia l'uso di Spice che le relazioni analitiche che sovrintendono al dimensionamento dei vari circuiti.

NOTA PER IL LETTORE

La simulazione al computer è stata eseguita col programma SPICE (*Simulation Program with Integrated Circuit Emphasis* della Microcode Engineering Inc. (CircuitMaker).

ITIS "LEONARDO DA VINCI" PISA

a cura del Servizio per l'Orientamento *(itipisa@tin.it)*

Le origini dell'Istituto Tecnico Industriale di Pisa risalgono agli albori del regno d'Italia, quando nel 1871 fu fondata la "Scuola Tecnico Industriale" per iniziativa del Comune di Pisa. Nel 1886 fu divisa in due Istituti: la "Scuola Industriale" sostenuta ancora dal Comune e la "Scuola Tecnica" sovvenzionata dallo Stato. Dopo varie trasformazioni, nel 1908 la Scuola fu riordinata e posta alle dipendenze del Ministero Agricoltura, Industria e Commercio.

Una bella immagine aerea dell'Itis "Leonardo da Vinci" di Pisa

Nel1915 passò sotto il Ministero dell'Educazione Nazionale e fu classificata come "Regia Scuola Industriale di 2° grado". Dopo il riordinamento del 1924, divenne "Regia Scuola di Tirocinio".

Nel 1926, infine, fu riordinata come "Regio Istituto Tecnico Industriale" assumendo la struttura base che mantiene tuttora.

L'I.T.I.S. "Leonardo da Vinci" si trova a Pisa in via Contessa Matilde, la strada che corre di fianco alle mura di Piazza dei Miracoli. L'enorme complesso si sviluppa su una superficie continua di circa 43.000 mq dei quali più di 9.000 mq risultano coperti da edifici destinati ad aule, laboratori, officine, palestre, bar.

Possiamo dire che i numerosi laboratori, utilizzati durante tutti i cinque anni dei corsi di studio e di seguito elencati, rappresentano il piatto forte dell'Istituto e ognuno di essi è dotato di apparecchiature tradizionali e moderne, sufficienti per consentire agli studenti un'esperienza diretta sugli strumenti, singolarmente o in piccoli gruppi. La funzione del laboratorio è molto importante: i concetti un po' complicati possono essere osservati, toccati con mano, e così diventano un po' meno complicati; una volta appresa la teoria, possiamo applicarla per piccoli progetti, ed i progetti possono essere realizzati e verificati mediante le numerose apparecchiature presenti. La formazione di un tecnico non può fare a meno dei laboratori, ed è per questo motivo che la scuola destina buona parte delle risorse al loro continuo aggiornamento; rispetto alla continua evoluzione tecnologica in certi settori, le risorse non sono ancora sufficienti, ma l'esperienza acquisita in laboratorio consente dopo un breve periodo di addestramento, di utilizzare anche le apparecchiature e gli strumenti più evoluti.

INTERVISTA AL DIRIGENTE SCOLASTICO, ING. CORRADO MANFREDONIA

Da quanto tempo dirige questa scuola?

Ho il piacere di essere qui all'ITIS dal 1983.

Quali sono gli indirizzi di studio?

Abbiamo sei indirizzi di studio: chimica fisica ambientale e sanitaria, costruzioni aeronautiche, edilizia, elettronica e telecomunicazione, elettrotecnica e automazione, meccanica.

Come si vede un ventaglio di possibilità molto ampio.

Quali sono le differenze sostanziali tra un istituto tecnico, un professionale e un liceo?

La differenza tra il titolo rilasciato dalla nostra scuola e quello delle altre, è che questo permette, dopo il tirocinio e l'esame di stato, l'iscrizione ad un albo professionale e quindi di eserci-

tare la libera professione ed essere imprenditori di noi stessi.

I programmi di studio sono attuali?
La scuola nel suo complesso e molti dei nostri insegnanti collaborano con il mondo industriale. L'analisi dei sistemi , quello scolastico e quello industriale spinge l'Istituto ad un continuo aggiornamento sia per quanto attiene le conoscenze, le discipline e le nuove tecnologie.

I vostri contatti con l'esterno?
I contatti con le attività industriali, gli Enti pubblici zonali e l'Università, sono costanti e proficui. I ragazzi nelle classi terminali dei corsi di studio svolgono periodi di stage. Ciò permette di ridurre lo scollamento che di solito esiste tra sistema scolastico e mondo imprenditoriale.
L'Istituto lavora in stretto contatto con l'Università di Pisa nella realizzazione dei corsi SSIS (scuola di specializzazione post-laurea per l'insegnamento) discipline di meccanica ed elettronica.
Inoltre l'Istituto è accreditato presso la Regione come agenzia formativa per lo svolgimento di corsi di formazione e post diploma in sinergia con Comuni e Amministrazioni Provinciali.

Ci indichi tre buoni motivi per iscriversi a questa scuola.
Ottima preparazione dei diplomati spendibile sia per ulteriori approfondimenti universitari, che per intraprendere attività professionali sia subordinate che autonome. Elevata richiesta di periti industriali sia nel mercato del lavoro zonale che nazionale. E, non per ultimo, ottimo clima nei rapporti personali tra allievi, docenti e collaboratori.

Figura 1: L'orologio solare dell'ITIS

ALCUNI DATI SULL'ISTITUTO

- 43.000 metri quadri di spazio.
- 9.000 metri quadri coperti.
- Tanto verde e giardini.
- Parcheggio interno per auto e motocicli.
- Bar-ristoro.
- Biblioteca rinnovata.
- Sala lettura con postazioni multimediali.
- Aula musicale.
- 2 palestre attrezzate.
- Sala pesi.
- Aula magna con postazione multimediale.
- Corsi di Judo e ginnastica nelle palestre.
- Museo degli strumenti.
- 35 laboratori di cui 11 d'informatica.
- 750 alunni.
- 35 classi.
- 101 Docenti.
- 1 Dirigente Scolastico.
- 1 Direttore Servizi Amministrativi.
- 12 Assistenti Tecnici.
- 9 Assistenti Amministrativi.
- 16 Collaboratori Scolastici.
- 1 Servizio biblioteca.

PROGETTUALITÀ E CORSI DI FORMAZIONE DELL'ISTITUTO

Progetti di Orientamento alle scelte, destinati agli studenti delle classi seconde finali con interventi di vario genere, interni ed esterni all'Istituto, finalizzati alla lettura delle proprie abilità, capacità e aspettative.

Progetti IFTS - istruzione tecnico scientifica superiore Progetti POST-DIPLOMA, destinati ad utenza diplomata (anche in altri Istituti). L'obiettivo principale di questi interventi è la formazione di tecnici con elevata specializzazione particolarmente richiesta dal mercato del lavoro.

Progetti di Alfabetizzazione e di Formazione Lavoro, destinati a drop-out, ragazzi in obbligo formativo, disoccupati, inoccupati, extracomunitari; interventi per l'acquisizione di competenze tecnico professionali nell'ottica di prevenzione della disoccupazione e di inserimento lavorativo attraverso la riconversione o l'acquisizione dei saperi/abilità.

Progetti di Percorsi Pilota finalizzati alla definizione di profili professionali diversi dagli standard.

Attività di volo, destinate ai ragazzi del corso di Costruzioni Aeronautiche (come attività propedeutica alla conoscenza della strumentazione di bordo ed alla elaborazione di un piano di volo).

Attività sportive: preparazione e partecipazione a manifestazioni provinciali, regionali, nazionali.

Partecipazioni a gare Olimpiche Culturali (fisica, chimica, matematica).

Preparazione al conseguimento della Patente Europea del Computer (E.C.D.L).

Figura 2: L'aereo costruito dalla sezione aeronautici

Figura 3: Uno dei tanti laboratori di informatica

IL PROGETTO: COME GESTIRE UN MOTORE PASSO-PASSO CON WINDOWS®

Dei Proff. Domenico Iracà, Stefano Mazzantini, Elio Tomaselli, Manola Pisani e degli alunni Polino Antonio, Zaccagnini Francesco, Bettinelli Stefano
itipisa@tin.it

Gestione di un motore passo-passo da personal computer con programma di gestione in ambiente Windows attraverso la porta parallela Centronics con rilevazione della posizione e della velocità dell'asse con encoder incrementale.

Il presente progetto è stato proposto come unità didattica e come esercizio da completare ad un gruppo di allievi della classe 4 Telecomunicazioni sezione B nell'anno scolastico 2002-2003.
Il progetto è relativo al controllo ad anello aperto di un motore passo-passo collegato con una interfaccia realizzata ad hoc connessa ad un PC attraverso la porta parallela centronics. Il programma di controllo è stato implementato in linguaggio C++ utilizzando il compilatore VisualC++6.00 e la tecnologia Microsoft Foundation Classes (MFC).
Ovviamente, trattandosi di una applicazione didattica non ci sono grosse pretese di precisione nella misura e nella impostazione della velocità del motore; del resto essendo il motore controllato direttamente da un programma Windows ed essendo quest'ultimo un sistema NON real-time, i tempi di attivazione e di acquisizione da e sull'I/O non sono assolutamente isocroni, cioè possono variare in modo impredicibile.
Il motore passo-passo costituisce uno spunto didattico interessante per la sua semplicità, per la sua reperibilità e per la facilità con la quale può essere comandato in modo digitale. Inoltre esso si presta bene ai primi approcci con l'elettronica dei BJT in quanto sono richiesti soltanto transistori che lavorano in ON-OFF, cioè in saturazione-interdizione, con bassissima dissipazione di potenza, e questo è come noto, un concetto che viene assimilato abbastanza facilmente e bene dagli allievi.

L'unità didattica è stata così suddivisa:

1. Sono stati esposti brevemente il funzionamento e la struttura di un motore passo-passo a 4 fasi prelevato da una vecchia stampante.
2. È stata esposta la tecnica di misura della velocità e della posizione angolare per mezzo di un encoder ottico incrementale.
3. È stata esposta la tecnica di costruzione di alcuni semplici programmi con interfaccia visuale utilizzanti la tecnologia MFC.
4. È stato realizzato un programma di controllo appositamente studiato per pilotare il motore in questione.
5. È stata realizzato lo schema elettrico di una interfaccia in grado di fornire la necessaria potenza alle fasi del motore, non potendo ovviamente queste essere pilotate direttamente dalla porta parallela. Nello schema è inserito inoltre il circuito per il condizionamento del segnale fornito dall'encoder ottico.
6. È stato realizzato lo sbroglio della scheda per la successiva fase di costruzione del circuito stampato.
7. Il circuito stampato è stato fresato con una fresa per prototipi LPKF in dotazione all'Istituto.

Struttura e funzionamento di principio dei motori passo-passo.
La struttura di principio di un motore passo-passo (del tipo detto unipolare) si compone di un magnete permanente e di 4 solenoidi, detti FASI, disposti e collegati come in figura 1. Il magnete permanente coincide con il rotore del

motore mentre le espansioni polari delle 4 bobine costituiscono lo statore. L'albero è perpendicolare al piano del foglio e nella realtà è solidale con il magnete, passando per il centro di esso.
E' facile intuire che il magnete può essere trascinato in rotazione azionando in sequenza le fasi A, B, C, D. Supponiamo, ad esempio, che attivando la fase A nasca un NORD sulla espansione polare superiore. Il SUD del rotore sarà attratto da tale polarità e il rotore stesso ruoterà in senso orario in modo da ridurre la distanza che separa i due poli, trovando la posizione di equilibrio. Se ora spegniamo la fase A e attiviamo la fase B, il rotore compirà un quarto di giro in senso orario; spegnendo la fase B ed azionando la C, avremo un altro quarto di giro e così via.

Ci sono 3 modi per controllarne gli spostamenti: 2 sono detti a passo intero, uno a mezzo passo (esiste comunque anche il modo detto micro-stepping).
Nel modo a passo intero "full step normal drive mode", vengono attivati 2 avvolgimenti per volta, in questo caso mediante i buffer A e B consentendo, così, il passaggio di corrente negli avvolgimenti A e B. Si producono, in questo modo, i 2 poli nord che faranno ruotare il rotore di 90° in senso orario (il suo sud si fermerà al centro dei 2 nord di statore). Successivamente si azionano B e C, poi C e D infine D e A e abbiamo completato un giro. Questo modo di funzionare è quello che sviluppa la maggiore coppia e provoca il maggiore assorbimento di corrente.

Naturalmente invertendo la sequenza temporale delle attivazioni dei buffer il rotore ruoterebbe in senso contrario.

Un altro modo di controllo a passo intero (detto "full step wave drive mode") si ottiene attivando un solo avvolgimento per volta alternando gli statori da magnetizzare. Ad esempio se attiviamo dapprima il buffer A, producendo un solo polo nord in alto mediante l'avvolgimento A, il rotore si ferma col suo sud in alto; se, poi, attiviamo il buffer B producendo un polo nord a destra mediante l'avvolgimento B il rotore ruota di 90° in senso orario per fermarsi col suo sud a destra. Si farà concludere il giro attivando prima il buffer C, poi il D.
Anche questa volta invertendo la sequenza (buffer D-C-B-A) si inverte il senso di rotazione.
Questo modo di controllo sviluppa meno coppia e dà minore assorbimento di corrente.

Il terzo modo è detto "half step mode": in esso vengono alternati i 2 modi di controllo: 1 passo in wave drive mode, il passo successivo in normal drive mode, e così via, il rotore si sposta solo di 45° ad ogni passo anziché 90.
Questo modo di controllo sviluppa una coppia variabile, dà anche un assorbimento variabile di corrente, ma consente dei posizionamenti più precisi dei dispositivi meccanici mossi dal motore.

Ovviamente il senso degli avvolgimenti deve essere lo stesso per tutte le 4 bobine, per far sì che per un medesimo verso della corrente che circola in ognuna di esse si abbia la stessa polarità rivolta verso il rotore. Nella pratica 4 capi omologhi delle bobine vengono riuniti e collegati a comune all'alimentazione, e gli altri 4 capi omologhi vengono azionati secondo quanto descritto per mezzo di interruttori elettronici a BJT. È interessante comunque fare qualche prova "a mano" che permette di accorgerci ad esempio che il motore passo-passo, a differenza dei motori DC o AC, può stare fermo con coppia (cioè sta fermo ma è bloccato).

Vale appena la pena di rilevare che la velocità di rotazione del motore può essere variata semplicemente

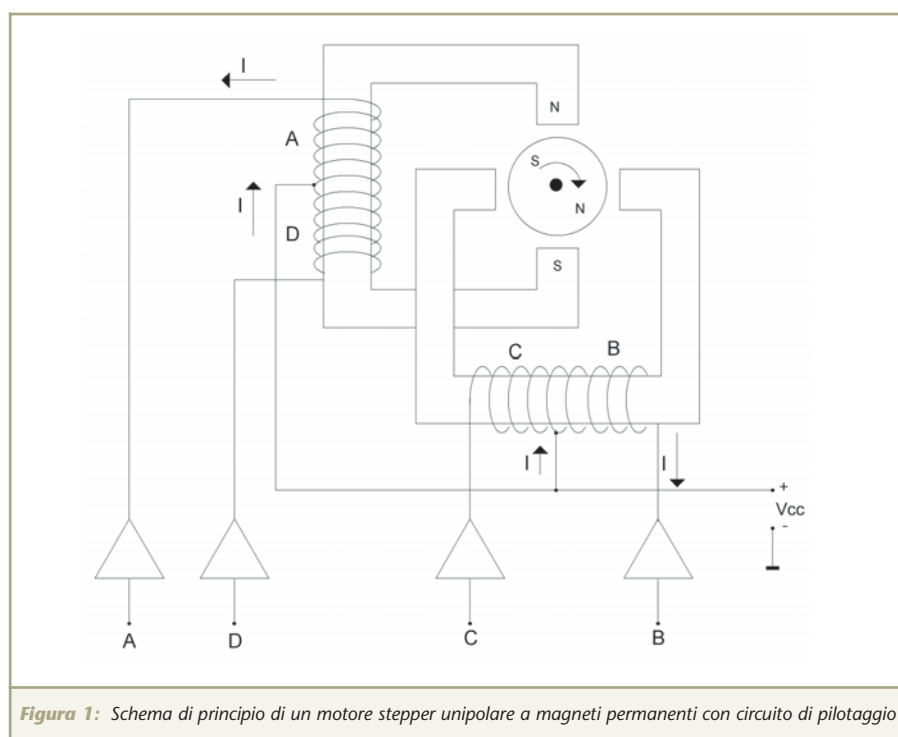



Figura 1: *Schema di principio di un motore stepper unipolare a magneti permanenti con circuito di pilotaggio*

variando il tempo che intercorre fra le attivazioni delle varie fasi. Pertanto la velocità di rotazione può andare da 0 fino ad un valore che è semplicemente dettato dalla inerzia meccanica del motore, che ad un certo punto non riesce più a seguire il campo magnetico rotante e "perde il passo" (non possiamo andare sotto qualche millisecondo di attivazione fra le fasi).
Rileviamo infine che un motore reale compie molti più passi dei 4 (o otto) consentiti dalla struttura di principio, in quanto il rotore è realizzato in modo da avere diverse decine di coppie Nord-Sud.
Inoltre esiste un altro tipo di motore, detto bipolare, nel quale la corrente negli avvolgimenti cambia senso e richiede una elettronica di pilotaggio più complicata.

In figura 2 è riportato un semplice circuito di azionamento delle 4 fasi per il pilotaggio unipolare. I diodi, detti di ricircolazione o free-wheeling, servono a cortocircuitare le extra correnti di apertura delle bobine quando i BJT passano dalla interdizione alla saturazione e devono essere dimensionati in modo da dissipare l'energia magnetica che si accumula nelle bobine quando i BJT conducono.

In definitiva quindi il funzionamento (per ottenere un giro) nei vari modi del motore unipolare sarà riassunto dalle seguenti tabelle (si suppone che il senso crescente corrisponda al senso orario di rotazione: scandendo la tabella in senso decrescente avremo il senso antiorario):

FULL STEP NORMAL MODE

Passo \ Fase	A	B	C	D	Hex
1	1	1	0	0	0x0C
2	0	1	1	0	0x06
3	0	0	1	1	0x03
4	1	0	0	1	0x09

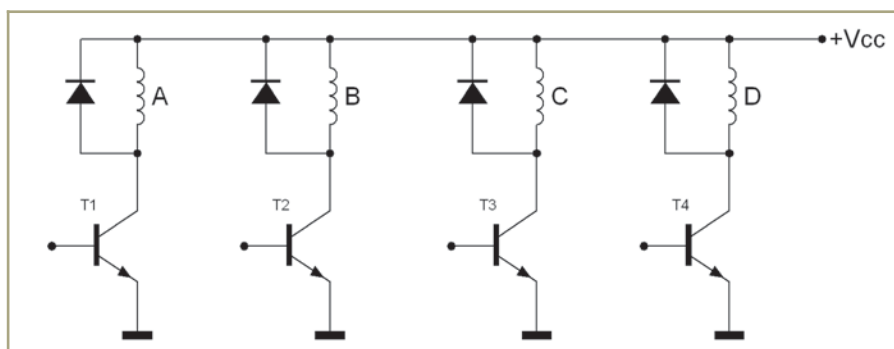



Figura 2: Schema di montaggio dei transistor e delle fasi del motore passo-passo

FULL STEP WAVE MODE

Passo \ Fase	A	B	C	D	Hex
1	1	0	0	0	0x08
2	0	1	0	0	0x04
3	0	0	1	0	0x02
4	0	0	0	1	0x01

FULL STEP WAVE MODE

Passo \ Fase	A	B	C	D	Hex
1	1	0	0	0	0x08
2	1	1	0	0	0x0C
3	0	1	0	0	0x04
4	0	1	1	0	0x06
5	0	0	1	0	0x02
6	0	0	1	1	0x03
7	0	0	0	1	0x01
8	1	0	0	1	0x09

Misura della posizione e della velocità angolare di un albero motore con encoder incrementale.
Un encoder incrementale è un dispositivo molto semplice che a seconda della precisione con cui viene realizzato può costare da pochi centesimi fino a diverse centinaia di euro. Qualsiasi mouse ne contiene due, uno che rileva gli spostamenti in orizzontale e uno per gli spostamenti in verticale. Si possono recuperare anche dalle stampanti, in genere abbinati al motore di trascinamento del carrello. Esso è costituito da un disco forato, solidale all'albero del quale si vuol misurare posizione angolare e/o velocità, posto fra un emettitore di luce infrarossa e un fotodiodo o fototransistore. L'emettitore è costantemente acceso e il ricevitore produrrà quindi un numero di impulsi al secondo pari al numero di giri moltiplicato per il numero di fori dell'encoder. È ovvio che la precisione (e quindi il costo) dell'encoder dipenderanno appunto dal numero di fori per una data dimensione. Nello schema di figura 3 il trigger di Schmitt 74HC14 è necessario per "pulire" la forma d'onda che è presente sul collettore del BjT e che per vari motivi non è una forma d'onda rettangolare idonea ad essere processata dagli stadi TTL successivi.
Gli encoder sono usatissimi perché molto robusti, semplici, affidabili e insensibili ad interferenze ambientali di vario tipo. Alcuni sono provvisti di due serie di fori sfalsati di un quarto di passo; ogni serie dispone di una propria coppia emettitore-rivelatore; in tal modo è possibile riconoscere la direzione, a seconda che i fronti di un segnale siano in anticipo o in ritardo rispetto a quelli dell'altro. Infine alcuni tipi hanno un foro di riferimento che serve a

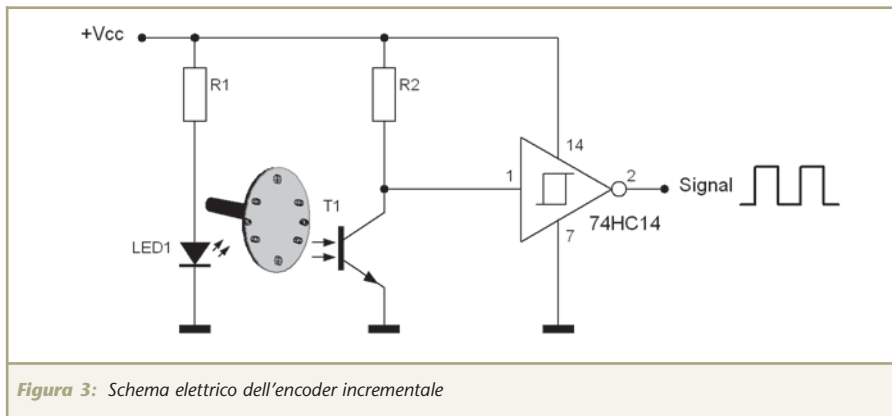



Figura 3: Schema elettrico dell'encoder incrementale

conoscere con certezza quando si è all'inizio di un giro. La precisione di questi dispositivi può essere molto elevata, arrivando, in alcuni tipi fotoincisi, anche a 2048 impulsi per giro su corone di un paio di cm di diametro.
La posizione angolare può essere misurata semplicemente come:

$$\alpha = \frac{(360 * N_impulsi_totali)}{N_FORI} \bmod 360$$

Il numero di giri totali come:

N_impulsi_totali / N_FORI

Essendo il N_impulsi_totali il numero di impulsi dall'accensione del motore e ottenuto accumulando ogni T_misura il numero N_impulsi contato in detto tempo.
La velocità in giri al minuto come:

$$60* \frac{N_impulsinT_misura}{N_fori*T_misura}$$

CREAZIONE DEL PROGETTO SOFTWARE PER WINDOWS 9X UTILIZZANDO LE MICROSOFT FOUNDATION CLASS DI VC++5/6

Nota: il docente si è limitato a fornire uno scheletro di programma e le spiegazioni di seguito riportate; le successive modifiche e aggiunte sono opera degli studenti; di conseguenza alcune cose possono apparire diverse nel software definitivo.
Obiettivo: realizzare un programma Windows in grado di comandare un motore passo-passo mediante l'interfaccia parallela della stampante e nel contempo di visualizzare la posizione dell'albero mediante conteggio dei fori di un encoder ottico incrementale. La scelta della porta parallela della stampante come interfaccia, è dettata dal fatto che tale porta è stata "dimenticata" nel meccanismo di accesso privilegiato all'I/O nelle versioni Windows 3x-9x; per altre porte le cose non sono così "semplici", purtroppo, e occorrono tecniche speciali e librerie DLL proprietarie per gestire questo meccanismo.

Alcune brevi considerazioni su Windows e sulla programmazione ad oggetti

Il nome Windows deriva dal fatto che tutti gli elementi del sistema operativo, sia quelli visibili, che altri nascosti, appartengono ad una **classe di oggetti** che si chiamano windows. L'utente del sistema può interagire in vario modo con pulsanti, icone, caselle di testo e altri elementi che vengono chiamati in gergo controlli, ma che sono per il sistema operativo altrettante finestre. Una finestra può in genere contenere altre finestre.
Windows è un sistema operativo orientato ai **messaggi**, vale a dire tutto è regolato da messaggi in forma di identificatori numerici e dati che partono dai controlli di una finestra e arrivano ai controlli di un'altra finestra. Non è detto che tutti gli oggetti di windows siano visibili.
Esiste in Windows una compenetrazione profonda fra i concetti di finestra, oggetti e classi di oggetti. Infatti le finestre sono oggetti che possiedono **proprietà** (es. backcolor: colore di sfondo), funzioni dedicate alla loro manipolazione (cioè **metodi**, es: show() rende visibile una finestra o hide() che la rende invisibile) e **gestori degli eventi**, cioè particolari funzioni che vanno in esecuzione quando nella finestra accade un particolare evento (es: OnClick(): quando viene premuto un pulsante).
Ogni finestra eredita i metodi generali ed è possibile comunque aggiungerne, così come è possibile creare nuovi tipi (classi) di oggetti.
La differenziazione fra un metodo M applicato ad un oggetto O1 e lo stesso metodo applicato ad un oggetto O2 viene effettuato con la cosiddetta **notazione punto** (analoga a quella utilizzata per le strutture del C e del Pascal): O1.M e O2.M.
Se gli oggetti (o le proprietà) appartengono poi a classi diverse C1 e C2 avremo C1::O1.M e C2::O2.M. Da queste affermazioni discende che il linguaggio di elezione per creare applicazioni Windows è il C++, ovvero C con classi, dedicato alla programmazione ad oggetti (OOP) ed in particolare il VisualC++ dotato di classi "prefabbricate" relative a tutti gli oggetti di uso comune, minimizzando la necessità di creare nuove classi. Ciò non toglie che potrebbe essere utilizzato qualsiasi altro linguaggio, ad esempio anche l'assembler, ma tutto sarebbe molto più faticoso. Intercettando i vari messaggi che le finestre si scambia-

no si possono ottenere i comportamenti necessari per i nostri scopi.
I programmi hanno una finestra principale che processa i messaggi che arrivano dalle altre finestre. Quando si preme un pulsante in una finestra, dalla finestra che rappresenta il pulsante parte un messaggio verso la finestra del programma; quando tale messaggio arriva, viene processato. In realtà questo sistema è molto lento, per questo Windows **non è assolutamente un sistema in tempo reale** e, per applicazioni specifiche, è inadeguato: non è infatti garantito il tempo di arrivo dei messaggi perché esso dipende fortemente da quanti ce ne sono in giro e da quanto sono veloci i processi che li generano e che li usano…pertanto non è garantito il tempo di risposta ad un evento. Possiamo dire che "forse" l'evento verrà processato.
Un sistema operativo a finestre in tempo reale molto robusto è ad esempio **QNX**.
Il compilatore VisualC++ crea una struttura (detta appunto FRAMEWORK) in cui praticamente tutti i messaggi relativi ai possibili eventi sono già processati con delle funzioni dette **default event handler** (gestore predefinito dell'evento).
Questa struttura tipicamente è uno scheletro vuoto che genera una o più finestre di dialogo con eventualmente pulsanti e menù di default, alle quali però non è ancora associata alcuna azione. E' nostra responsabilità riempire questo template, simulacro, con qualcosa di sensato. Con un tool del compilatore (il class-wizard), possiamo scegliere di modificare la gestione predefinita sostituendo il gestore predefinito con un nuovo gestore fatto da noi (è quello che facciamo quando con il class-wizard clicckiamo nella message map su un evento…).
Il class-wizard ci permette di scegliere il messaggio al quale vogliamo associare un override-handler (gestore modificato rispetto al predefinito). Volendo possiamo inserire nel codice una nostra gestione, prima o dopo quella predefinita. Ad esempio, selezionando il messaggio **WM_INITDIALOG** che il framework invia quando una finestra viene creata e inizializzata:

```
/* questa è la routine del
framework; oltre al lavoro
predefinito farà ora altro */
BOOL
CSTEPPERDlg::OnInitDialog()
{
   /* qui mettiamo il codice
che ci serve… */

   /* questa è la chiamata al
gestore predefinito della clas-
se base che verrebbe invocata
quando la finestra viene crea-
ta */
CDialog::OnInitDialog();

   /* qui mettiamo il codice
che ci serve… */

return TRUE;
   /* ritorna TRUE se non si
sposta il focus su un control-
lo */
}
```

Questa tecnica viene chiamata **OVERRIDING** (da to override - scavalcare), perché scavalca il comportamento predefinito. Se il nostro codice non esiste si avrà il comportamento predefinito.
Un'altra cosa importante da ricordare è che le nostre classi rappresentano con strutture dati tipo STRUCT, con metodi e proprietà, gli oggetti con cui lavoriamo e possono essere generate dal class-wizard a partire da classi già esistenti dette classe basi ereditando in tal modo tutte le proprietà e i metodi delle classi base.
Una cosa che può tornare utile è la MessageBox: essa può essere utilizzata per far comparire delle finestre di messaggio e quindi anche per delle rapide verifiche.
Se è usata in una funzione statica, non membro di una classe va scritta come :

```
MessageBox(NULL, buff, "messag-
gio di configurazione:",
MB_ICONINFORMATION);
```

Il nostro programma sarà sviluppato come una applicazione **multithreaded**.
Un applicazione multithreaded (thread - filone) è un programma suddiviso in più filoni di esecuzione parallela che cooperano per elaborare dati comuni. Un particolare componente del sistema operativo fa avanzare un thread alla volta per pochi millisecondi. In tal modo possono essere portate avanti più attività in parallelo, anche se è un parallelismo simulato.
Nel nostro caso questo è necessario perché i gestori degli eventi delle finestre non possono essere bloccanti, se non per pochissimo tempo (qualche millisecondo) pena il malfunzionamento o il non funzionamento dei controlli; non esiste un metodo main che possa tollerare un ciclo while infinito, cosa che invece è necessaria per azionare il motore e per controllare il sensore.
Una soluzione possibile, ma molto complicata, soprattutto con Windows, è lavorare ad interrupt; una soluzione più semplice è creare un thread principale responsabile della creazione e dell'aggiornamento delle finestre e un thread secondario responsabile della ricezione dei caratteri. Questo thread secondario può stare in attesa attiva, bloccante, senza alcun danno per il thread principale e per la funzionalità delle finestre.

I passi da seguire per la realizzazio-

ne del software sono i seguenti:

1 Aprire **mscdev** (il compilatore).
2 Fare click su **FILE>NEW>PROJECTS>MFC application (exe)**. Viene avviata una autocomposizione (o wizard - mago) nella quale occorre specificare un nome per il progetto (viene creata una cartella contenente tutti i file del progetto, eventualmente è possibile specificare anche il percorso della cartella). Scegliere ad esempio **STEPPER**.
3 Rispondere alle varie richieste: **dialog-based, no activeX controls, no about-box** e lasciare tutto il resto come di default. Alla fine dopo aver premuto FINISH avremo un riassunto del nostro progetto da riempire.
4 Fare click su **BUILD> build stepper.exe** (ci dovrebbe essere un tasto di scorciatoia sulla toolbar…).
5 Fare click su **BUILD> execute stepper.exe** (ci dovrebbe essere un tasto di scorciatoia sulla toolbar…). A questo punto avremo una dialog box che potremo modificare. Chiudiamola con **OK**. Il wizard ci ha creato uno "scheletro" vuoto; è nostra responsabilità riempirlo come ci è più utile.
6 La struttura del nostro progetto sarà alla fine la seguente: avremo un **task principale** che si occuperà della gestione degli **input** ed eventualmente degli **output** da e sulla **finestra di dialogo**, un **task secondario** che si occuperà della **gestione delle fasi** del motore e **un altro task secondario** che si occuperà del controllo del **trasduttore di posizione** encoder incrementale. La comunicazione fra i tre task avverrà ad esempio per mezzo di **variabili globali o meglio, se possibile, di variabili di classe**.
7 Avremo bisogno dei seguenti input:
  a **direzione** oraria o antioraria, utilizzando dei **radio-button**
  b **modo** half o full step, sempre con dei radio-button
  c **start e stop**, ancora con dei radio-button
  d **velocità** con una **casella di testo** in sola lettura e un **controllo spin**, o anche un **controllo slider** (e magari una progress bar)
  e un **pulsante** per resettare il contatore della posizione dell'encoder incrementale
  f una **list-box** per selezionare la porta parallela del PC
8 Avremo bisogno dei seguenti

output:

a Una **casella di testo** in sola lettura per visualizzare la posizione dell'encoder incrementale.
b Una **casella di testo** in sola lettura per visualizzare la velocità dell'encoder incrementale.
c Una **casella di testo** in sola lettura per visualizzare il numero di giri dell'encoder incrementale.

9 Modifichiamo la nostra finestra di dialogo per inserire i nostri elementi dei punti 7 e 8: andiamo sulla linguetta **RESOURCES** e clicchiamo su **STEPPER RESOURCES>DIALOG> IDD_STEPPER_DIALOG**. Si aprirà un editore grafico con il quale potremo inserire, eliminare, modificare i nostri controlli. Ingrandiamo alquanto la nostra finestra, rendendola di dimensione circa 4 volte maggiori (si può comunque modificare in ogni momento). Eliminiamo la scritta Fare: disponi… facendoci click sopra e poi **edit‡cut**. Se non compaiono i controlli in una toolbar fare click con il tasto destro in una area grigia dei menu e selezionare **controls**. Se non compare qualcosa fare click su **view‡**qualcosa.

a Inserimento **radio button**: selezionare il controllo radio button nella toolbar e trascinarlo nella dialog-box. Ripetere per quanti radio-button vogliamo.
b Inseriamo **3 caselle di testo** cliccando sul controllo casella di testo nella toolbar e trascinandolo dove ci pare più opportuno nella dialog box.
c Inseriamo **un bottone** cliccando sul controllo button e trascinandolo dove ci pare opportuno nella dialog box.
d Inseriamo **un controllo slider** cliccando sul controllo slider è possibile fare in modo che non appena questo controllo viene spostato vada in esecuzione una routine (gestore dell'evento, event-handler) che aggiorna una variabile del programma.
e Inseriamo una **progress-bar** cliccando sul controllo progress bar e trascinando. La progress bar ci indicherà in modo visivo la velocità impostata; essa verrà aggiornata dallo stesso event-handler dello slider.
f Inseriamo **dei commenti** (static label) dove ci pare accanto ai controlli per chiarire a cosa servono. I commenti inizialmente si chiamano tutti static.

10 Adesso dobbiamo fare alcune modifiche ai controlli per personalizzarli e renderli idonei ad "appenderci" del codice eseguibile. Facciamo click con il tasto destro del mouse e facciamo comparire il menu a tendina nel quale clicchiamo "**properties**". Quando compare la finestra proprietà clicchiamo sullo spillone per fare in modo che la finestra non si richiuda. Adesso possiamo modificare tutte le proprietà dei nostri oggetti inseriti nella dialog box e volendo le proprietà della dialog box stessa, ad esempio il titolo (caption), i font o altro.
Attenzione: gli **ID_** o **IDC_** o **IDD_** (a sinistra nella finestra proprietà) sono dei simboli che rappresentano dei numeri con cui viene fatto riferimento agli oggetti nelle varie parti del programma; non ce ne devono ovviamente essere due o più uguali, ma comunque si possono anche scegliere di fantasia. Tutti i controlli possono essere ridimensionati e/o allineati fra loro e con riferimento ad una griglia mediante mouse e/o **LAYOUT** nel menù principale. Dovrebbe essere abbastanza facile, almeno dopo alcuni tentativi, ottenere una struttura come quella di figura 4.

11 Miglioriamo ancora la struttura: Le due caselle di testo vengono rese **read-only** (cercare in proprietà come si fa); rinominiamo gli ID delle caselle di testo IDC_POSIZIONE, IDC_ACTSPEED e IDC_GIRI; impostiamo la proprietà number per queste caselle di testo, rinominiamo l'ID dello slider come IDC_SPEED.
Occorre infine sistemare i radio button. I **radio button** possono essere raggruppati e si può fare in modo che attivandone uno si disattivino gli altri. Più esattamente dobbiamo impostare per tutti l'attributo **auto** e sul primo del gruppo di controlli che vogliamo rendere interdipendenti l'attributo **group**. I controlli in una dialog-box sono ordinabili in modo crescente (vedremo, con layoutfl> tab order);

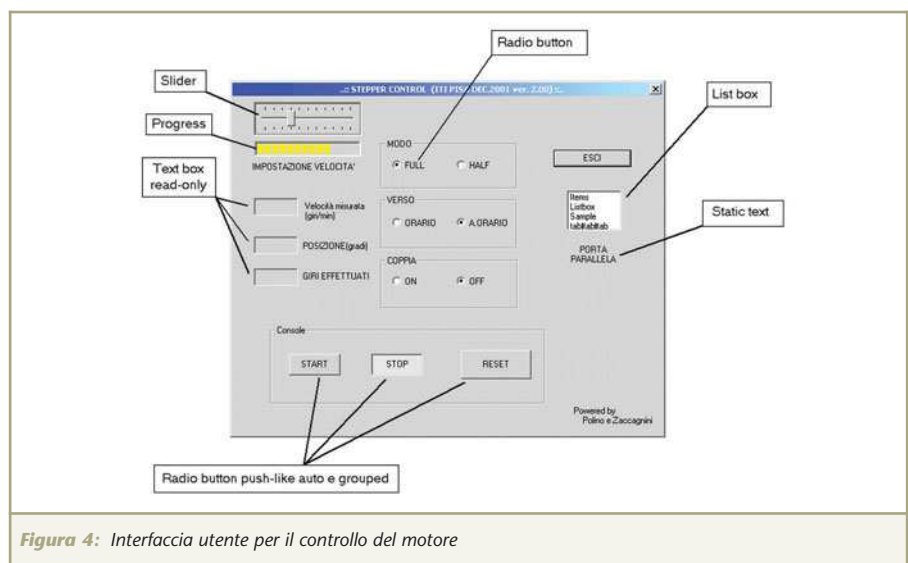



Figura 4: Interfaccia utente per il controllo del motore

costituiscono un **gruppo** i controlli con ordinale consecutivo compresi fra un controllo con group impostato e l'ultimo con **group** non impostato. Occorre fare in modo che nel gruppo modo cliccando su full si disattivi half e viceversa e analogamente per **ORARIO** e **A.ORARIO**. Per far questo impostare la **proprietà GROUP** di **FULL** e di **ORARIO** e assicurarsi che i bottoni abbiano la proprietà **AUTO** impostata e che **FULL**, **HALF** e **ORARIO**, **A.ORARIO** siano in **tab order rispettivamente consecutivi**. Testare con **LAYOUT>TEST**. Se tutto è ok non ci devono essere interazioni fra i controlli di gruppi diversi. Già che ci siamo possiamo, solo per nostra comodità rinominare gli ID dei radio button con dei nomi più ovvi di quelli proposti di default: **IDC_RADIO1>IDC_FULL** e così via… sono solo nomi.
Possiamo inoltre conferire ai radio button start e stop l'aspetto di "pulsante interbloccato".

#define IDC_FULL	1000
#define IDC_HALF	1001
#define IDC_CW	1002
#define IDC_CCW	1003
#define IDC_POSIZIONE	1005
#define ID_SPEED	1006
#define IDC_RESET	1008
#define IDC_START	1009
#define IDC_STOP	1010
#define IDC_PORTA	1011
#define IDC_COPPIAON	1012
#define IDC_COPPIAOFF	1013
#define IDC_ACTSPEED	1016
#define IDC_GIRI	1019
#define IDC_BAR	1020
#define IDC_SPEED	1022

Tabella 1: ID dei controlli del progetto

Adesso salviamo tutto e chiudiamo la finestra di editing delle dialog box e delle risorse.
La tabella 1 riporta gli ID dei controlli aggiunti al progetto (la tabella fa parte del file resources.h).

12 Adesso aggiungiamo delle variabili (dette di classe) che dovranno rispecchiare lo stato dei controlli. Clicchiamo sulla linguetta FILES del workspace. Viene mostrata la lista delle cartelle che contengono i files che costituiscono il nostro progetto, in particolare i .h e i.cpp. Quello sul quale lavoreremo è **stepperDlg.cpp**, che è il file responsabile della gestione della finestra di dialogo (pr questo c'è il suffisso Dlg). Facciamo click con il tasto destro del mouse nella finestra di destra e nel menu che si apre clicchiamo su **CLASSWIZARD**.
Operiamo come segue:

a Selezioniamo la classe **CStepperDlg** se non lo è già. Selezioniamo la linguetta **MEMBER VARIABLES**. Aggiungiamo le variabili di classe facendo doppio click sul nome dei vari controlli o selezionandoli e ciccando su add variable. Viene chiesto il nome (viene suggerito un prefisso m_, è bene lasciarlo per evitare confusioni future con altre variabili) e il tipo della variabile. Effettuare le impostazioni in modo da ottenere uno schema simile a quello riportato in figura 5.
Occorre fare attenzione a scegliere il tipo corretto della variabile che deve essere aggiunta fra quelli suggeriti quando si preme add variable.

b Selezioniamo ora la linguetta **Messages map**. Eliminiamo, per snellire il codice, i metodi OnPaint e OnQueryDragIcon selezionandoli e ciccando delete. Questo elimina in realtà solo i prototipi, dobbiamo poi in stepperDlg.cpp cercare i corpi ed eliminarli a mano, come viene suggerito dopo aver fatto delete. Adesso dobbiamo creare i gestori degli eventi relativi alla pressione del pulsante e del controllo spin (o slider).
Per fare questo nella finestra messages map selezionare ad

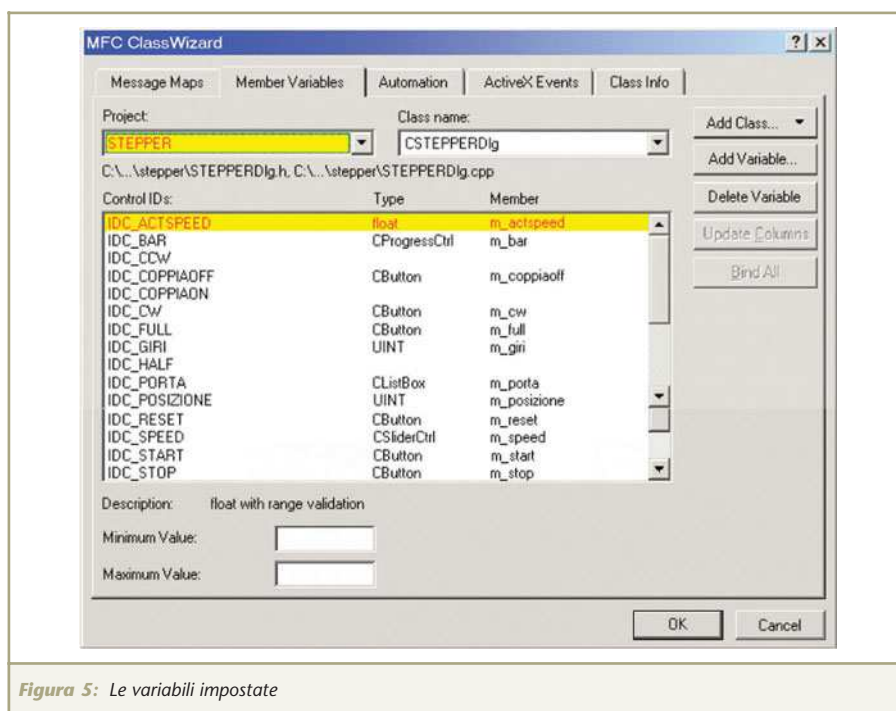



Figura 5: Le variabili impostate

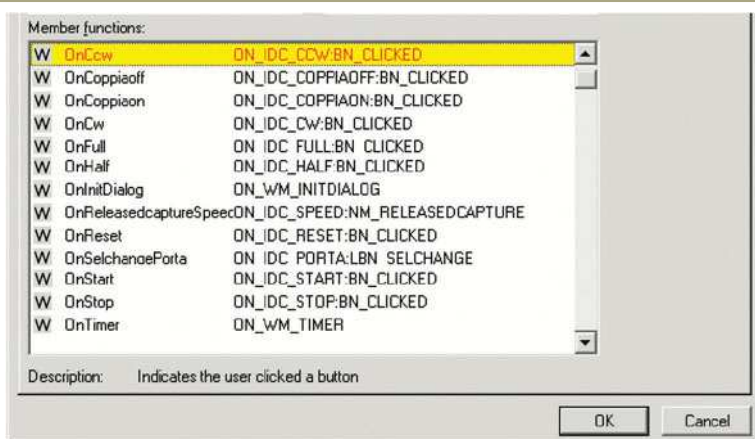



Figura 6: I gestori di evento

esempio **IDC_RESET** e fare doppio click su **BN_CLICKED**; viene suggerito il nome **OnReset** per la routine di gestione. Accettare. Ripetere per il controllo **IDC_SPEED**, selezionando **OnReleasedCapture**. I tasti OK, CANCEL hanno già una loro gestione di default che ci può andar bene, quindi li lasciamo perdere.

Complessivamente con questa tecnica aggiungeremo i metodi (in verità gestori di evento).

13 Adesso possiamo cominciare a scrivere il codice. Notiamo intanto che qualche controllo ha, come risultato dei punti precedenti, una variabile di classe associata scelta di un tipo utile a semplificare il lavoro di aggiornamento del controllo (lettura e/o visualizzazione). Ogni variabile di classe è una struttura avente proprietà e funzioni (metodi) che possono essere utilizzati per operare su di esse mediante la notazione punto (dot notation). Ad esempio una cosa utile è modificare il range di default dello spin button che va da 100 a 0 nel range da 200 a 400 mediante l'inserimento dell'impostazione:

m_speed.SetRange(200, 400);

Dove? Nella funzione dall'ovvio nome **init_dialog** che il compilatore crea per noi!

14 Quello che dobbiamo fare adesso è creare due task o thread, come detto uno per la gestione delle fasi (solito sistema: sleep e outp) e uno per la lettura del sensore (aspetto in loop che si abbia una transizione…). Questi processi operano leggendo e scrivendo da e su variabili (globali e/o di classe) che vengono ricopiate da e nei controlli dalla funzione DoDataExchange creata dal wizard. Il wizard stesso provvede anche ad inizializzare le varibili di classe che abbiamo associato ai controlli a seconda di ciò che abbiamo scritto dentro in properties.

I task si creano con la chiamata della funzione:

```
AfxBeginThread(nome_funzione_ch
e_implementa_il_task, puntato-
re_a_parametro);
```

La funzione che implementa il task deve essere di tipo con ritorno UINT (intero senza segno) e normalmente è in ciclo senza fine. I thread vengono uccisi automaticamente quando muo-re la finestra che li genera.

Mettiamo la creazione dei thread nella solita procedura init_dialog, responsabile della inizializzazione della finestra. Purtroppo in VC++ i task non appartengono alla stessa classe CstepperDlg delle variabili m_ e non hanno accesso diretto alle stesse (ricordare che i membri di una classe non hanno accesso diretto ai membri di una altra classe e meno che mai una funzione che non fa parte di una classe ha accesso a membri di una classe).

Una cosa importante quindi è capire come i due task possano comunicare fra loro e con il task principale che gestisce la finestra. Esistono due

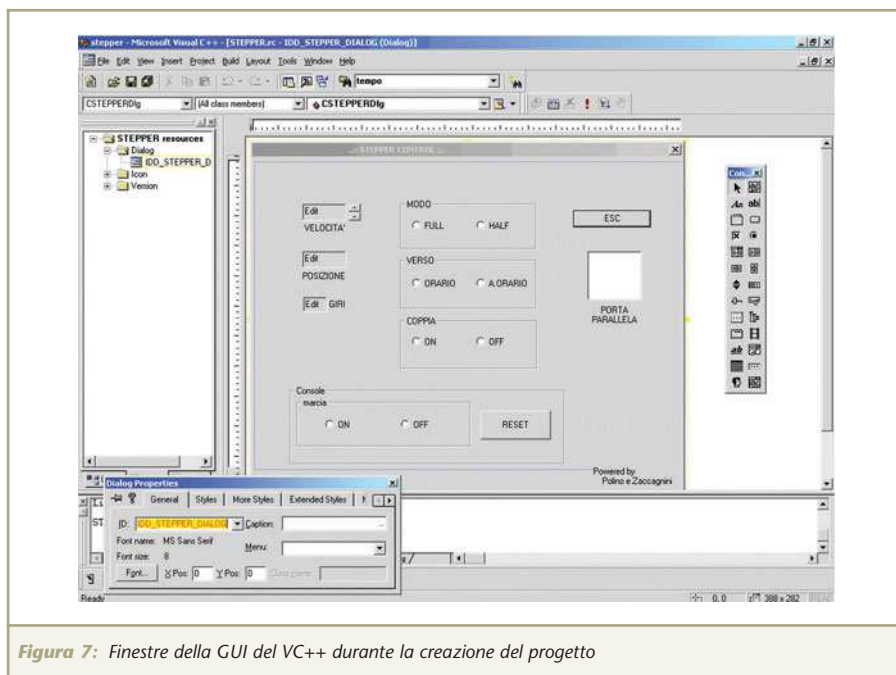



Figura 7: Finestre della GUI del VC++ durante la creazione del progetto

sistemi: uno “sporco” ma più efficiente dal punto di vista della velocità di esecuzione, che utilizza le variabili globali, e uno “pulito” ma meno efficiente che passa alla funzione che implementa il task un puntatore all’oggetto (in realtà alla zona di memoria) che rappresenta la finestra e tutti i suoi controlli. Questo secondo metodo lavora nel modo seguente: il task viene creato con un comando del tipo:

```
AfxBeginThread(fasi,
(LPVOID)this);
```

This è un puntatore all’oggetto corrente, cioè la nostra finestra. Nel task si può accedere, sia in lettura che in scrittura, agli oggetti di una finestra di tipo, ad esempio appunto, CSTEPPERDlg con una scrittura del tipo:

```
UINT fasi(LPVOID pt)
{
   /* creiamo un puntatore a
oggetto CSTEPPERDlg inizializzato all'indirizzo della area
di memoria //relativa alla
finestra corrente */

CSTEPPERDlg * p = (CSTEPPERDlg
*)pt;

   /* ora con p->oggetto possiamo accedere a qualsiasi
oggetto, metodo o proprietà
della finestra //CSTEPPERDlg:
esempio */

if(p->m_cw.GetCheck())....
// altro....
}
```

Purtroppo il sistema è pulito ma non efficiente perchè tanto tempo di esecuzione viene perso nei dereferenziamenti dei puntatori; pertanto questa soluzione non è utilizzata in quanto provando si è visto che il motore non può essere controllato con precisione che a velocità molto basse.
Usando invece le variabili globali abbiamo però due problematiche

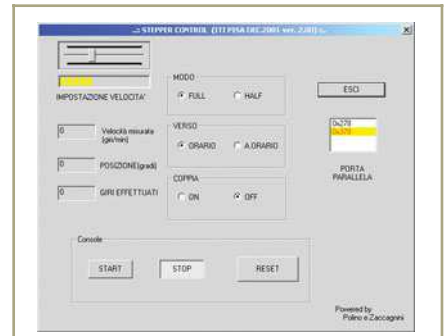



Figura 8: Schema Versione definitiva della interfaccia grafica: la velocità è impostata con uno slider e viene mostrata la velocità rilevata con l’encoder

diverse:

- **a** Le modifiche dell’utente sui controlli (ad esempio il cambio di stato di un bottone) sono comunicate al task per mezzo del corrispondente cambio di stato di una variabile globale che è manovrata dall’event handler del controllo: esempio: un radio-button premuto fa sì che vada in esecuzione il relativo event-handler che aggiorna variabile globale e allora il task modifica la sua azione in rela-

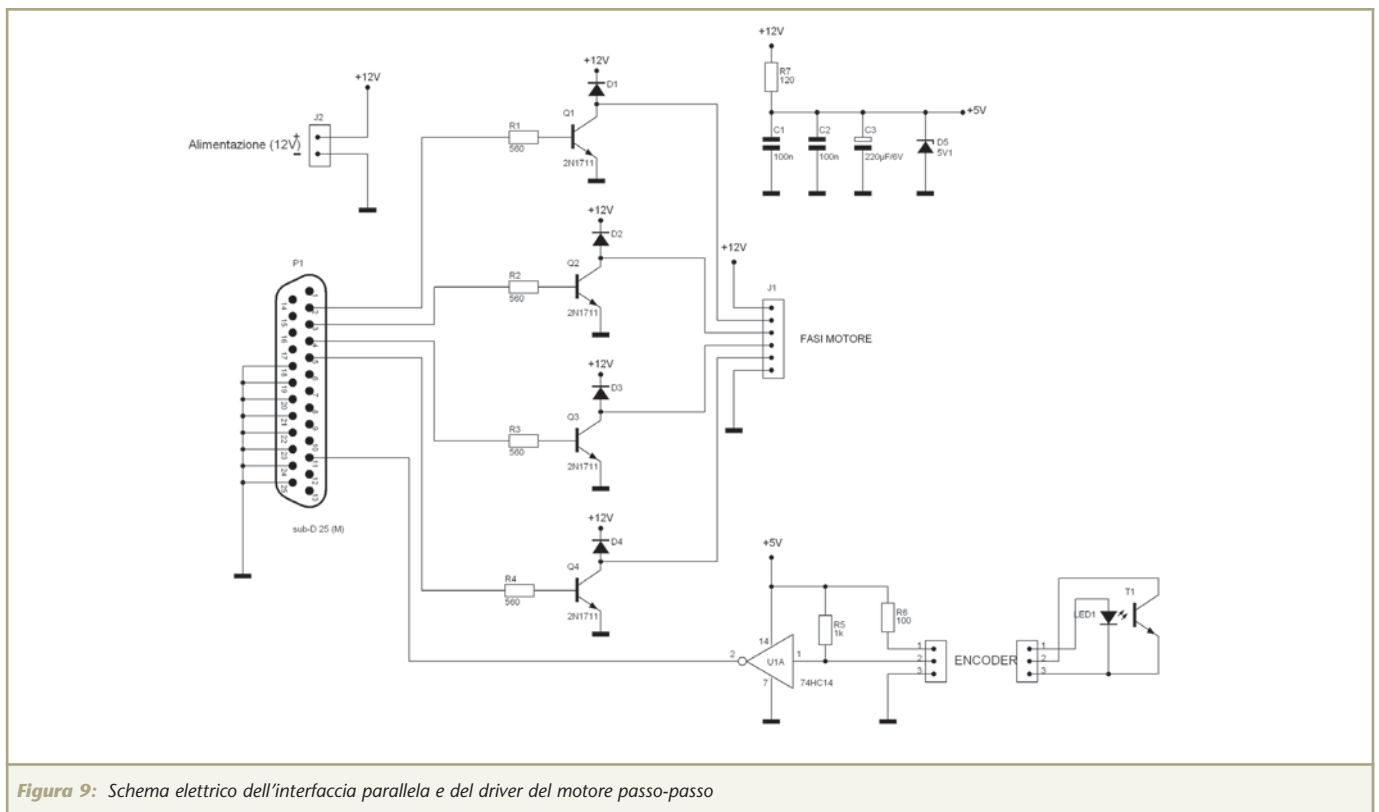



Figura 9: Schema elettrico dell’interfaccia parallela e del driver del motore passo-passo

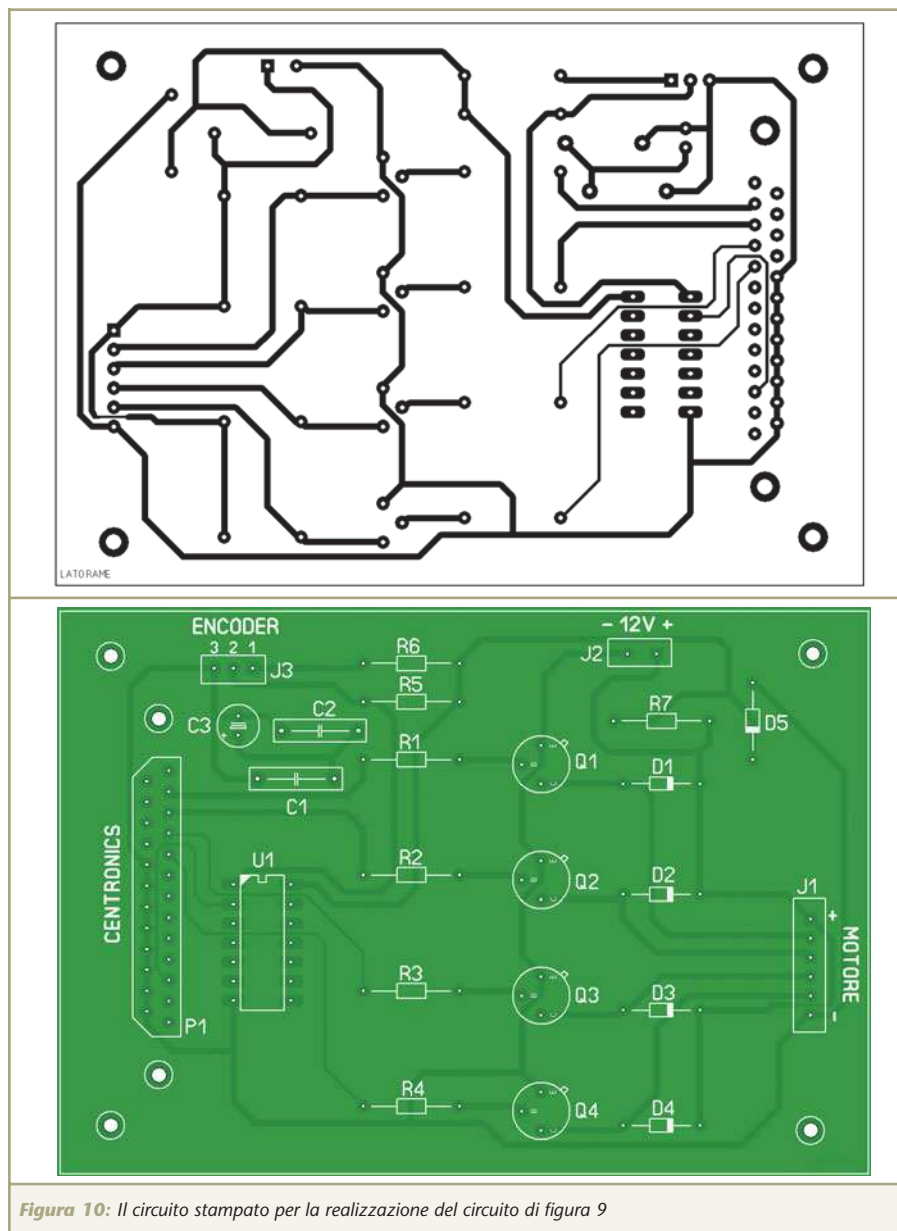



Figura 10: Il circuito stampato per la realizzazione del circuito di figura 9

Elenco componenti	
Sigla	**Valore**
Q1÷Q4	Transistor 2N1711
R1÷R4	560 Ω 1/4 W
R5	1 KΩ 1/4 W
R6	100 Ω 1/4 W
R7	120 Ω 1/2 W
C1,C2	100 nF ceramico
C4	200 µF 6 V elettrolitico
D1÷D4	Diodi 1N4001
D5	Diodo Zener 5,1 V 1/4 W
U1	Integrato 74HC14
P1	Connettore Centronics (maschio da cs)

zione al nuovo stato della variabile globale.

b Viceversa se il task deve aggiornare un controllo allora esso aggiornerà una variabile globale; periodicamente lo stato di tutte le variabili globali destinate a questo tipo di aggiornamento verrà ricopiato nella variabile di classe a supporto del controllo (o anche direttamente nel controllo con SetDlgItemText) da un ulteriore event-handler della classe detto **OnTimer** che ogni tot millisecondi effettua la ricopiatura.

Essendo l'utente finale un uomo, lento, il sistema va bene. Il timer serve anche, nel nostro caso per implementare il tempo di misura per il calcolo della velocità, in base a quanto visto prima per l'encoder. Per creare un processo timer si può, sempre nella init_dialog utilizzare la funzione SetTimer(*numero_identificativo_timer, tempo_millisecondi, NULL*); che crea un timer che scade ogni tmpo_millisecondi.
Quando il timer scade si può effettuare le operazioni necessarie.
Per creare la funzione di gestione del timer dobbiamo andare nel class wizard e nella finestra Messages map relativamente alla classe CstepperDlg cercare il messaggio WM_TIMER e fare doppio click su di essa accettando il nome proposto. Dopo di che potremo fare edit code e andarla a riempire con il codice:

```
void CSTEPPERDlg::OnTimer(UINT
nIDEvent) // ogni T_MISURA:
{
   float tmp;

   if(flag) m_actspeed =
(60000 * impulsi) / (NFORI *
T_MISURA);  // velocità
   impulsi = 0;  flag = false;
   tmp = ((float)posizione /
NFORI); m_giri = tmp;  // giri
   m_posizione = ((UINT)(360 *
tmp) % 360);  //posizione;

   CDialog::OnTimer(nIDEvent);
// questa è la chiamata dell'-
handler di default generata
dal wizard
}  // non rimuoverla…
```

15 Possiamo ora compilare il tutto dopo le necessarie modifiche per ottenere il funzionamento globale.

Se volete informazioni su un metodo o su una proprietà potete mettere il cursore all'inizio della proprietà stessa e poi premere F1: ad esempio per m_cw.SetCheck(false); mettersi dopo il punto. Se volete velocemente informazioni su una intera classe, ad esempio CButton, scrivere Cbutton ovunque nel file, andare prima della C e premere F1; guardare class members. Ovviamente poi ricancellare Cbutton.

SCHEMA ELETTRICO INTERFACCIA DI POTENZA

Come si vede dallo schema di figura 9, le fasi del motore risultano collegate ai bit 0,1,2,3 della porta parallela di indirizzo porta, (pin 2,3,4,5 del DB25); il sensore è invece collegato al bit 7 della porta di indirizzo porta+1 pin 11) del DB25. Lo stepper utilizzato è stato recuperato da una vecchia stampante; esso è dotato di 6 fili di raggruppati in 2 gruppi di 3: bianco-rosso-verde e giallo-blu-verde; è evidente quindi che i due fili verdi sono i capi comuni delle 4 fasi e possono essere uniti e connessi alla alimentazione positiva (12 o 24 Volt).
Dopo alcune prove a mano si è poi verificato che il senso di attivazione degli altri capi per la rotazione in senso orario in full-step è giallo-bianco-blu-rosso.
Come si evince dai dati di targa ogni fase del motore ha una resistenza interna di 40 Ohm, pertanto, volendo alimentarlo a 12V ogni BJT deve sopportare una corrente pari a IL = IC = 12/40 = 300 mA; un 2N1711 è perfettamente adatto allo scopo e non necessita nemmeno di radiatore; per quanto riguarda la resistenza di base, ipotizzando un hFE minimo (con il BJT in saturazione) di 100 otteniamo:

$$R_B = \frac{(V_{CH} - V_{BEsat})\, h_{FEsat}}{I_L} = \frac{(2.4-0.8)100}{0.3} = 530 \text{ ohm}$$

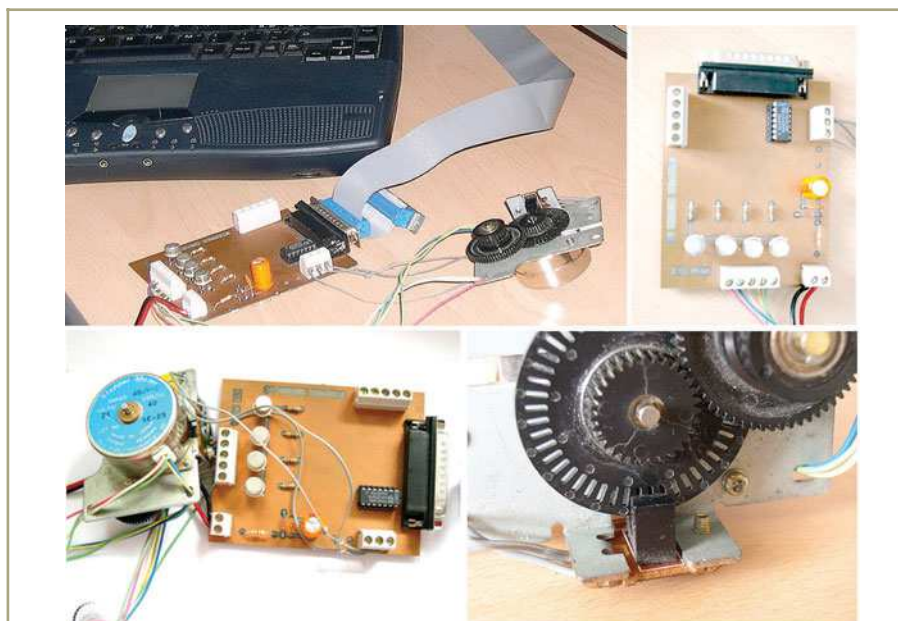



Figura 11: Fotografie della scheda di interfaccia finita e del sistema complessivo

IL PROGRAMMA DI GESTIONE SUL PC

Il programma di gestione è disponibile sul sito di Fare Elettronica (www.farelettronica.com) e, nel formato sorgente, sono indicate le modifiche effettuate rispetto allo scheletro prodotto dal wizard del compilatore. L'unico file dello scheletro che va modificato a mano è STEPPERDlg.cpp.

I modi di funzionamento implementati sono:

- Full step wave-mode.
- Half step.
- Senso orario.
- Senso antiorario.
- Motore fermo con coppia (acceso con scansione delle fasi bloccata).
- Motore fermo senza coppia (spento)

Figura 12: Gli autori del progetto: i ragazzi dell'ITIS di Pisa con i docenti Proff. Tomaselli, Mazzantini, Iracà

CAMPAGNA ABBONAMENTI

2004-2005

Abbonarsi a Fare Elettronica significa ricevere, comodamente ogni mese a casa tua, tante idee e consigli per rendere il tuo hobby una vera passione.

Perché Fare Elettronica si diverte solo quando ti diverti tu.

Abbonandoti potrai ricevere la rivista ad un prezzo molto interessante rispetto a quello di copertina. Pagherai infatti solo € 39 invece di € 51 con un risparmio di ben € 12 oltre a ricevere uno **sconto del 10% su tutti i libri del catalogo SANDIT (più di 800 titoli).**

1 ANNO
11 RIVISTE
A SOLI € 39

IL BUS CAN

di Nicola Ulivieri
nicola@ing.unisi.it

*Dopo il bus 1-Wire, presentato nel numero precedente, parliamo questo mese di un altro bus che utilizza due soli fili come supporto fisico per la trasmissione e che offre grandi potenzialità. Il Controller Area Network (**CAN**) è un bus il cui impiego è ormai consolidato in ambito industriale. Il suo costo, prestazioni e facilità di aggiornamento forniscono una eccezionale flessibilità nel progetto dei sistemi.*

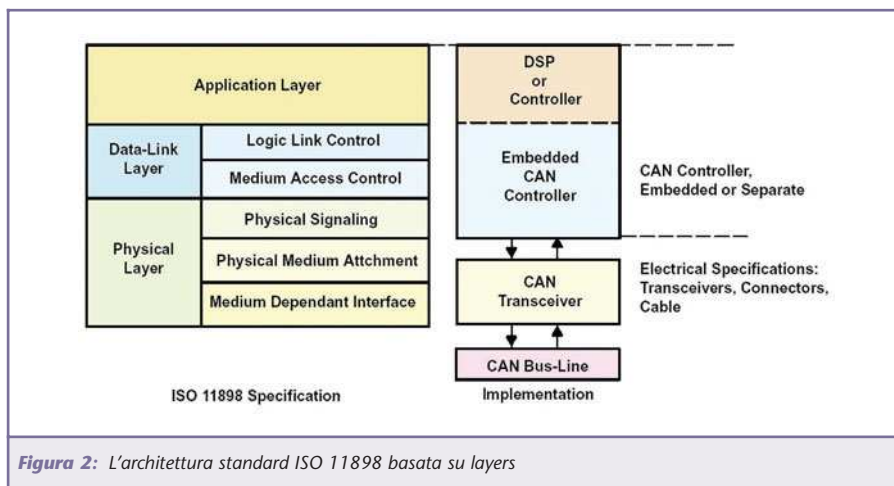
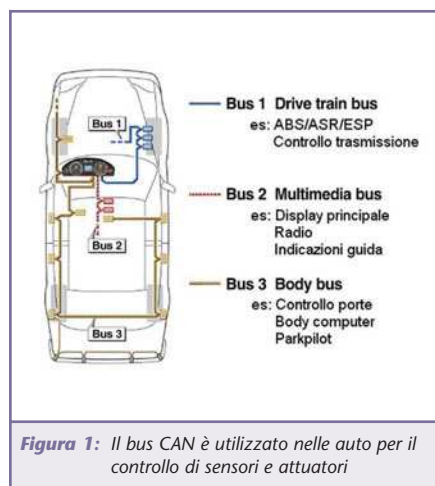
Il bus CAN è una rete seriale originariamente creata per l'industria automobilistica ma che si è presto affermata sia nell'automazione industriale sia in molti altri campi. Il Controller Area Network è definito come un sistema multi-master che invia messaggi in broadcast ("a tutti i dispositivi connessi") con velocità massima pari a 1M bit al secondo (Mbps). A differenza delle reti USB o Ethernet, ad esempio, il CAN non utilizza lunghi blocchi di dati per le trasmissioni da un nodo ad un altro (trasmissione point-to-point) sotto la supervisione di un dispositivo Master. Nel CAN, invece, messaggi brevi come la comunicazione di una temperatura o la velocità (RPM), sono spedite in broadcast a tutta la rete.

LO STANDARD CAN

Il CAN è un bus di comunicazione seriale definito da uno standard ISO (International Standardization Organization), sviluppato per l'industria automobilistica con lo scopo di rimpiazzare il complesso groviglio di cavi con due soli fili, permettendo così la semplificazione della stesura dei fili elettrici e consentendo un notevole risparmio. Le specifiche dichiarano una velocità che può raggiungere 1Mbps, alta immunità alle interferenze elettriche e ottima capacità di auto-diagnosi e riparazione automatica degli errori. Queste sono le caratteristiche che hanno portato al successo di questo standard e al suo ampio impiego in vari campi quali, oltre al settore automobilistico già citato, quello aerospaziale, manifatturiero, marittimo ed anche in campo medico.

Il protocollo di comunicazione ISO11898 definisce come le informazioni devono essere passate tra i vari dispositivi sulla rete, conformemente al modello OSI (Open System Interconnection) definito in termini di layers. La comunicazione tra i dispositivi connessi al mezzo fisico è definita dal "*Physical layer*" del modello. L'architettura ISO11898 definisce i due layers inferiori dei sette layers del modello OSI/ISO, e cioè il *data-link layer il physical layer* (figura 2).

Figura 1: *Il bus CAN è utilizzato nelle auto per il controllo di sensori e attuatori*

Figura 2: *L'architettura standard ISO 11898 basata su layers*

L'Application layer stabilisce invece la comunicazione verso un protocollo per applicazioni specifiche, a livello superiore, come ad esempio il protocollo CANopen, indipendente dal venditore. Questo protocollo è supportato dal gruppo di costruttori e utenti internazionali CAN in Automation (CiA). Informazioni dettagliate possono essere reperite dal sito del CiA , www.can-cia.org. Ci sono inoltre molti altri simili protocolli emergenti, dedicati ad applicazioni specifiche come per l'industria dell'automazione o aviazione.
Esempi di protocolli detti industry-standard, basati sul CAN, sono *KVASER's CAN Kingdom, Allen-Bradley's DeviceNet e Honeywell's Smart Distributed System* (SDS)

STANDARD CAN O EXTENDED CAN? (QUESTO È IL PROBLEMA...)

Il protocollo di comunicazione CAN è un Carrier-Sense Multiple-Access con rilevazione delle collisioni e arbitraggio sulla priorità dei messaggi, in una parola **CSMA/CD+AMP**. CSMA vuol dire che ogni nodo deve osservare se ci sono segnali sul bus e aspettare un predefinito tempo di inattività prima di tentare di spedire un messaggio. CD+AMP sta a significare che le collisioni sono risolte attraverso un arbitraggio che si basa sulla priorità stabilita di ogni messaggio, indicata dall'identificatore del messaggio stesso.
L'identificatore con la priorità più alta vince sempre l'accesso al bus.

Figura 3: Sensore di pressione della Sensor-Technik dotato di interfaccia CAN

La prima versione del CAN, riportata in Tabella 1, ISO11519 (Low-Speed CAN) è stat ideata per applicazioni fino a 125 kbps ed utilizza un identificatore ad 11 bit. La seconda versione, ISO11898 (1993) ancora con identificatore ad 11 bit, incrementa la velocità

Nomenclatura	Standard	Massima velocità	Identifier
Low-Speed CAN	ISO 11519	125 kbps	11 bit
CAN 2.0A	ISO 11898:1993	1 Mbps	11 bit
CAN 2.0B	ISO 11898:1995	1 Mbps	29 bit

Tabella 1: Versioni del CAN

Nomenclatura	Descrizione
SOF	Segnala l'inizio del messaggio
Identifier	Stabilisce la priorità del messaggio. Più piccolo è il valore binario, maggiore è la priorità
RTR	Single Remote Trasmission Request. È un bit dominate quando l'informazione è richiesta da un altro nodo
IDE	Indica che l'identificatore è standard a 11 bit e non c'è estensione
r0	Riservato per usi futuri
DLC	Codice a 4 bit contenente il numero di bytes che verranno trasmessi
Data	Dati di lunghezza massima di 64 bit
CRC	Codice per la rilevazione di errori
ACK	Utilizzato per confermare la corretta ricezione o meno del messaggio
EOF	Indica la fine del messaggio
IFS	Indica il tempo richiesto dal controllore per per memorizzare correttamente il messaggio ricevuto

Tabella 2: I bit nei messaggi Standard CAN (figura 4)

C'era una volta...

La prima versione del bus CAN è stata sviluppata da Robert Bosch [2] nel 1986 quando gli fu commissionato lo sviluppo di un sistema per la comunicazione di tre ECUs (Electronic Control Units) poste in veicoli della Mercedes. Ormai la UART non era più adatta per queste esigenze perché consentiva la comunicazione tra due soli dispositivi (connesione point-to-point). La necessità di un sistema di comunicazione multi-master era ormai tassativa. Il primo chip CAN fu fabbricato nel 1987 da Intel.

fino a 1Mbps mentre il più recente emendamento ISO11898 (1995) introduce l'identificatore esteso a 29 bit. La seconda versione è spesso indicata come **Standard CAN Version 2.0A** mentre l'ultima è chiamata **Extended CAN Version 2.0B**. Per fare un po' più di confusione, il CAN standard è anche chiamato BasicCAN mentre c'è chi si riferisce alll'Extended con FullCAN. Lo standard CAN facente uso dell'identificatore a 11 bit (figura 4), fornisce quindi 2^{11}=2048 diversi identificatori per i messaggi mentre l'Extended CAN arriva a ben 2^{29} e cioè 537 milioni di diverse possibili combinazioni di identificatori.

Il CAN in breve (per chi non ha voglia...)

- Multi-master
- Protocollo di comunicazione CSMA/CD+AMP
- Basato su messaggi e non su indirizzi.
- Utilizza un cavo twisted-pair come supporto fisico per la trasmissione.
- Massima frequenza del segnale 1 Mbit/s
- Massima velocità di trasmissione dati (data-rate) effettiva 40 kBytes/s
- Massima latenza di messaggi ad alta priorità inferiore a 120 us a 1 Mbit/s
- Lunghezza del cavo massima 30m a 1Mbit/s; fino a 1km a velocità inferiori.
- Presenta alta affidabilità grazie ad un sofisticato controllo degli errori.
- Massimo numero di dispositivi connessi al bus elevatissimo in teoria, non superiore a 30 in pratica.
- Esistono due versioni principali:
  - 2.0A - Standard CAN - 11 bit Identifier
  - 2.0B - Extended CAN - 29 bit Identifier

MA COME FUNZIONA?

I messaggi contenenti dati, trasmessi da ogni nodo sul bus, non contengono indirizzi né del nodo trasmittente né di un possibile nodo ricevitore. Quello che viene fatto, invece, è etichettare il contenuto del messaggio con un identificatore (si, ancora lui) che è unico in tutta la rete. Tutti gli altri nodi (ah! nodo = dispositivo) sulla rete ricevono il messaggio ed eseguono un test di riconoscimento sull'identificatore per capire se il messaggio, e quindi il contenuto, è indirizzato a loro oppure no. Se lo è, viene processato, altrimenti sarà ignorato.

IL LAYER FISICO

Prima di fornire altri dettagli sul protocollo di comunicazione del CAN, vediamo come viene effettuata la trasmissione dei bit. Il layer fisico (figura 2) non è parte dello standard Borsch CAN ma è incluso nello Standard ISO11898. Nei messaggi CAN, i bit possono essere spediti come high (cioè 1), chiamato nel CAN anche bit Recessivo, oppure come low (cioè 0) che è invece il Dominante. Come indica il nome, lo 0 domina sull'1, che vuol dire che se un nodo tenta di spedire uno 0 ed un altro prova a spedire un 1, il risultato che si osserverà sul bus sarà uno 0 (figura 7). I segnali sono trasmessi su due fili chiamati CAN-High e CAN-Low che operano in modo differenziale (figura 8). Questo vuol dire che le tensioni che essi trasportano, variano in modo opposto sui due fili in modo da incrementare la tolleranza al rumore. Lo Standard ISO11898 prevede che per la trasmissione del bit Recessivo, entrambe le linee siano tenute a **2.5V** (linea continua nelle figure 7 e 8) mentre il bit Dominante viene trasmesso portando la linea CAN-High a **3.5V** e la linea CAN-Low a 1,5V

Figura 4: I bit nei messaggi CAN standard (o BasicCAN, identificatore a 11 bit) e CAN Esteso (o FullCAN, identificatore a 29 bit

Figura 5: Schede della National Instrument per l'interfacciamento al PC di reti CAN

Nomenclatura	Descrizione
>	I nomi non riportati hanno lo stesso significato dello Standard CAN
SRR	Sostituisce il RTR della prima versione
IDE	Indica che ci sono altri 18 bit identificatori che seguono
r1	Ulteriore bit riservato per usi futuri

Tabella 3: *I bit nei messaggi Extended CAN (figura 4)*

(doppia linea nelle figure 7 e 8), creando così una tensione differenziale tipica di 2V. "Notate che poichè per entrambe le linee del CAN sono definite delle tensioni positive, una domanda sorge spontanea: rispetto a cosa?
Sembra infatti esistere in letteratura un po' di omertà sul fatto che deve esistere un terzo conduttore per il riferimento di massa!"
Lo standard indica anche l'impedenza caratteristica della linea che deve essere 120Ω.

ARBITRAGGIO E IDENTIFICATORI

L'ormai famoso Identificatore determina anche la priorità del messaggio. La priorità più alta spetta all'identificatore con valore numerico più basso. Questo permette di stabilire quale nodo può accedere al bus se due di essi tentano di farlo contemporaneamente. Ogni dispositivo che trasmette sulla rete, osserva contemporaneamente lo stato del bus (figura 8). Questo perché, se tentando di spedire un 1, rilevasse invece uno 0 sulla rete, capirebbe allora di aver perso l'arbitraggio e devrà bloccare immediatamente la trasmissione per lasciare la precedenza al nodo con priorità più alta (figura 7).
In questo modo il messaggio a più alta priorità ha la certezza di venire trasmesso, proprio come se nessun altro tentasse di accedere al bus allo stesso tempo, evitando così la ritrasmissione dei messaggi che porterebbe ad una diminuzione delle prestazioni. I messaggi a priorità inferiore sono automaticamente ritrasmessi nel successivo ciclo oppure in quello dopo ancora, se altri messaggi a maggiore priorità continuano a venir spediti.

Non è consentito a due nodi della stessa rete di mandare messaggi con stesso ID. Se succedesse, l'arbitraggio non potrebbe funzionare. I nodi trasmittenti si accorge-

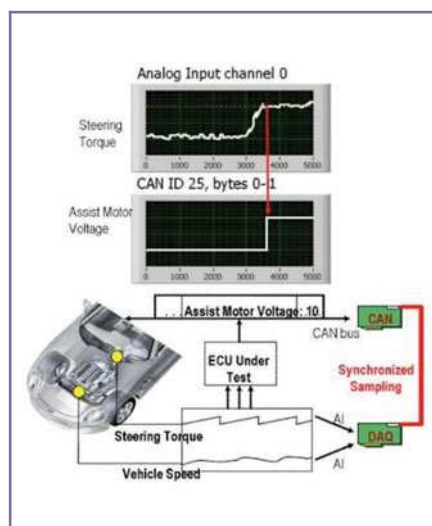



Figura 6: *Test dell'EPAS. In un servosterzo, o EPAS (Electronic Power Assist Steering System), un'unità di controllo elettronica acquisisce segnali, come la forza della sterzata e la velocità del veicolo, li passa ad un algoritmo di controllo e aggiusta i messaggi CAN che controllano il servomotore. Per validare l'algoritmo di controllo dell'EPAS, l'ingegnere dei test automobilistici può acquisire messaggi dal CAN e dalla scheda analogica (DAQ) simultaneamente, analizzarli e determinare se è stata prodotta l'uscita desiderata per il servomotore*

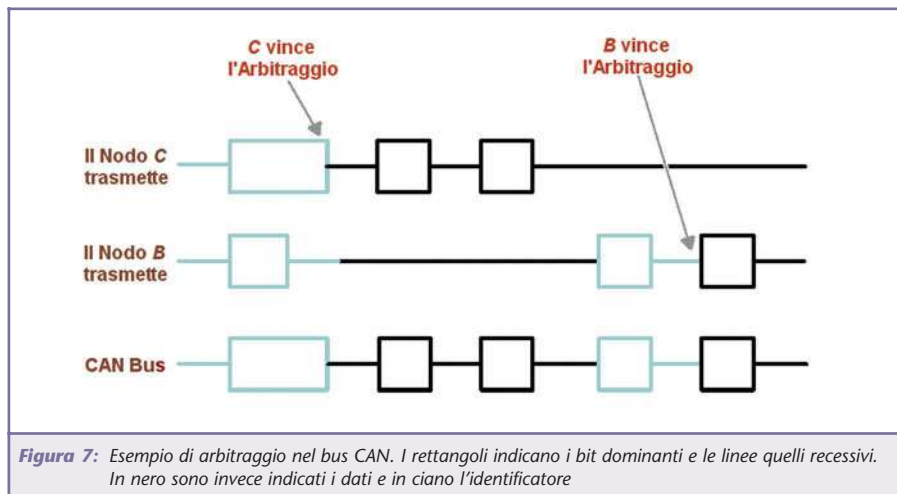



Figura 7: *Esempio di arbitraggio nel bus CAN. I rettangoli indicano i bit dominanti e le linee quelli recessivi. In nero sono invece indicati i dati e in ciano l'identificatore*

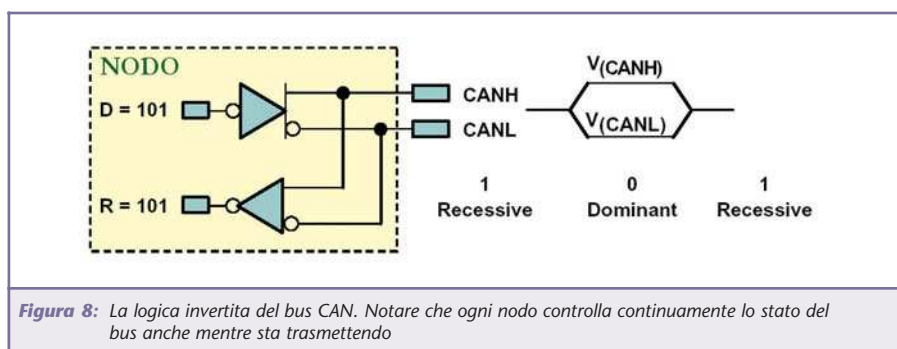



Figura 8: *La logica invertita del bus CAN. Notare che ogni nodo controlla continuamente lo stato del bus anche mentre sta trasmettendo*

rebbero che il loro messaggio viene distorto e quindi utilizzerebbero la correzione errori dello standard CAN che in caso di errori ripetuti porterebbe infine allo spegnimento automatico di uno dei nodi (bus-off mode).

TIPI DI DATI

Esistono quattro tipi principali di messaggi (o frames) nel CAN, chiamati *remote frames*, *error frames*, *overload frames* e *data frames*. Gli ultimi sono i messaggi utilizzati per la trasmissioni dei dati e sono i messaggi "normali" del CAN. I remote frames sono invece messaggi utilizzati per la richiesta di informazioni. Un frame con l'RTR alto (tabella 2, figura 4), vuol dire che il nodo trasmittente sta chiedendo informazioni del tipo specificato dall'identificatore. Il nodo con l'informazione disponibile dovrà rispondere spedendola sulla rete. In funzione dell'implementazione del controllore CAN, la risposta potrà essere spedita automaticamente. Controllori CAN più semplici (BasicCAN) non possono rispondere in automatico. In questo caso viene comunicata al microcontrollore host la richiesta e esso dovrà spedire il dato.
L'error frame è un messaggio speciale che viola le regole di formattazione di un messaggio CAN e viene trasmesso quando un nodo rileva un errore nel messaggio. L'overload frame, simile all'error frame, viene trasmesso da un dispositivo che è diventato troppo impegnato nelle trasmissioni e rischia di non poter gestire i messaggi nel tempo previsto.

Lunghezza del bus (m)	Velocità di trasmissione (kbps)
30	1000
100	500
250	250
500	125
1000	62.5

Tabella 4: Velocità massima dei segnali in relazione alla lunghezza dei cavi

GESTIONE DEGLI ERRORI ED ISOLAMENTO GUASTI

La robustezza del CAN può essere attribuita in parte alle sue abbondanti procedure per il controllo degli errori. Il protocollo CAN incorpora cinque metodi di controllo: tre a livello di messaggio e due a livello di bit. Se un messaggio fallisce uno di questi cinque controlli, non viene accettato e viene generato un error frame dal nodo ricevitore causando la ritrasmissione da parte del nodo trasmittente, finchè la ricezione non è corretta.
Comunque, se un nodo rileva un errore in continuazione, vuoi per disturbi, vuoi per un suo guasto, occuperebbe di conseguenza il bus all'infinito. Allora interviene il suo controllore che provvede a spegnerlo dopo un certo numero di tentativi di ricezione per liberare la rete.

CONCLUSIONI

Il CAN è un bus ideale per applicazioni che richiedono un elevato numero di brevi messaggi in un breve periodo di tempo. Inoltre, fornisce l'alta affidabilità richiesta per l'uso in ambienti operativi ostili. Poiché il CAN è basato su messaggi e non su indirizzi, è particolarmente adatto per grandi sistemi, quando i dati sono richiesti da più di una locazione e la loro integrità è obbligatoria.

Un altro rilevante vantaggio del CAN è la capacità di gestione automatica dei guasti. I nodi difettosi sono automaticamente sganciati dalla rete, evitando che tali dispositivi impediscano la comunicazione di tutto il bus, rendendolo così sempre disponibile per eventuali messaggi 'urgerti'.
Questo tipo di gestione degli errori permette anche l'aggiunta di dispositivi al bus mentre la rete sta funzionando, analogamente a quello che avevamo già visto per il bus 1-wire (funzionamento hot-swap oppure hot-plugging).

Terminiamo questo tutorial con una puntualizzazione sui compromessi a cui bisogna scendere nell'implementazione reale di una rete. La massima velocità di 1Mbps è ottenibile solo nei casi ideali in cui la rete ha una lunghezza non elevata e il numero di dispositivi connessi è basso.
Generalmente si usano bus di lunghezza inferiore a 30m con meno di 30 nodi connessi. Il data-rate effettivo che si raggiunge è 40kBytes/s.

BIBLIOGRAFIA

[1] Steve Corrigan, "Introduction to the Controller Area Network (CAN)", Application Report SLOA101, Texas Instrument web page: http://www.ti.com/
[2] http://www.can.bosch.com/
[3] http://www.ni.com/swf/presentation/us/can/
[4] http://www.can-cia.org/
[5] http://www.computer-solutions.co.uk/gendev/can-tech.htm

tutt'ELETTRONICA

BASTIA UMBRA

2004

(Perugia)

20/21 marzo

ore 9/19

elettronica • hardware • software
radiantismo • ricezione satellitare
telefonia • componenti
accessori • surplus • hobbistica
radio e dischi d'epoca

Expo Elettronica

Il grande Circuito di Fiere dell'elettronica & Co.

servizio gratuito!!

Per essere sempre aggiornati sulle fiere di elettronica inviate la vostra e-mail a info@exporadioelettronica.it, scrivendo nell'oggetto "iscrivimi news on line"

organizzazione **BLU NAUTILUS srl** • tel. 0541 439573 • www.blunautilus.it

Per ottenere un INGRESSO RIDOTTO scarica il biglietto dal sito www.blunautilus.it o presenta questa inserzione alla cassa

Rif. FARE ELETTRONICA

Le fiere e mostre mercato di Aprile

2004

3-4 Aprile 2004
A.B.C. DELL'ELETTRONICA

Dalla prima radiotrasmissione attraverso l'Atlantico, avvenuta nel 1901 ad opera di Guglielmo Marconi, sono nati telegrafo, radio, telefono, televisione, radar, telematica, Internet e tutto l'universo dell'informatica. A 103 anni da quell'esperimento, la possibilità di comunicare senza un supporto fisico (su tutto il pianeta o addirittura fino agli angoli più reconditi dello spazio siderale) si è evoluta in maniera prodigiosa e utilissima: dalle attività militari e della Protezione Civile, al mondo dell'informazione, al lavoro, al tempo libero.
ABC dell'Elettronica presenterà una gamma vastissima di prodotti: computer, software, periferiche, accessori e ricambi; telefonia fissa e mobile; video games; home entertainment (hi-fi, vhs, dvd, antenne, decoder satellitari e digitali); piccoli elettrodomestici e strumentazioni varie. Altrettanto assortito sarà il settore dell'elettronica «fai da te» con kit per auto costruzione, surplus, schede, circuiti; ricetrasmittenti Om, CB e componentistica per l'utilizzo radioamatoriale.

Luogo: Lario Fiere, Viale Resegone – ERBA (CO)
Orari: dalle 9.00 alle 18.00
Organizzatore: Blu Nautilus (www.blunautilus.it – Tel 0541.439573)
Ingresso: € 7,00 (il biglietto è valido anche per la Fiera dell'Astronomia)

3-4 Aprile 2004
FIERA DELL'ASTRONOMIA

Nell'ambito di ABC dell'Elettronica di svolge anche la Fiera dell'Astronomia, uno dei pochissimi momenti in Italia dedicati all'astronomia amatoriale. Accanto al lavoro degli scienziati, c'è un vasto pubblico che si dedica a questo affascinante hobby. Alla Fiera dell'Astronomia partecipano aziende e rivenditori di strumenti utili per l'osservazione e lo studio della vota celeste: semplici binocoli, potenti telescopi, accessori ed editoria specializzata: libri, riviste, corsi e software. Oltre all'esposizione commerciale, è l'occasione per incontrarsi fra "addetti ai lavori". Le Associazioni Astrofile e di Astronomi, partecipi in grande quantità, giocano un ruolo di primo piano, presentando la propria attività, materiali prodotti e immagini astronomiche, planetarie e di profondo cielo. E' perciò l'occasione giusta, anche per chi vuole avvicinarsi a questo argomento, per prendere contatti ed approfondire le proprie conoscenze in materia di cielo e stelle.

Luogo: Lario Fiere, Viale Resegone – ERBA (CO)
Orari: dalle 9.00 alle 18.00
Organizzatore: Blu Nautilus (www.blunautilus.it – Tel 0541.439573)
Ingresso: € 7,00 (il biglietto è valido anche per la Fiera ABC dell'Elettronica)

17-18 Aprile 2004
MOSTRA EMPOLESE DELLA RADIANTISTICA E DELL'ELETTRONICA

Mostra-mercato attrezzature elettroniche, radiantistiche, ricezione via satellite, informatica & editoria specializzata.

LUOGO: Palazzo Delle Eesposizioni – EMPOLI (FI)
ORARI: dalle 9.00 alle 18.00
ORGANIZZATORE: Studio Fulcro
www.studio-fulco - Tel 010.561111
INGRESSO: n.p.

17-18 Aprile 2004
FIERA MERCATO DELL'ELETTRONICA DI PESCARA

LUOGO: Fiera Adriatica – SILVI MARINA (TE)
ORARI: dalle 9.00 alle 18.00
ORGANIZZATORE: Studio di Camillo Antonino
(Tel 085.4215840)
INGRESSO: n.p.

24-25 Aprile 2004
2° ELETTRO@BIT

Dopo il successo dello scorso anno, la Gedit che la organizza e l'amministrazione Comunale che la ospita, hanno voluto confermare l'appuntamento per tutti gli appassionati del settore. Gli articoli presenti saranno: elettronica, informatica, radiantistica, componenti elettrici, radio d'epoca, telefonia, accessori, utensili per hobbysti.. Oltre 55 espositori provenienti da tutta la penisola hanno confermato la loro presenza alla seconda edizione della manifestazione.

LUOGO: Via Tirso – Zona industriale
CIVITAVECCHIA (ROMA)
ORARI: dalle 9:00 alle 13:00 e
dalle 15:00 alle 18:30
ORGANIZZATORE: GEDIT
www.sitopreferito.it/gedit - Tel 075.5991020
INGRESSO: € 3,00

24-25 Aprile 2004
39ª RADIOAMATORE / HI-FI

Una fiera che riunisce le ultime invenzioni nel campo della radiantistica, dell'elettronica e dell'informatica e dell'hi-fi, una vetrina per conoscere i progressi della tecnica e aggiornarsi sui nuovi sistemi di radiodiffusione, le ultime frontiere del car-audio, dell'audio/video e dell'emergente settore del car-theater.

LUOGO: Viale Treviso, 1 - PORDENONE
ORARI: dalle 9:00 alle 18:30
ORGANIZZATORE: Ente Autonomo Fiera Di Pordenone
www.fierapordenone.it – Tel 043.4232111
INGRESSO: € 7,50

VITAMINA C
settima parte

di Sergio Tanzilli
s.tanzilli@farelettronica.com

Continuiamo questo mese a fare esperimenti in C sul PIC 18F452. Useremo il compilatore PCH della CCS-C all'interno dell'ambiente di sviluppo standard Microchip MPLAB, trasferiremo il programma all'interno della memoria del PICmicro con il programmatore Microchip ICD2, proveremo il programma sulla scheda per esperimenti Microchip PICDEM 2 PLUS e le funzioni di debug a livello sorgente. Vedremo quindi come utilizzare un porta seriale del PICmicro per visualizzare i messaggi generati con la funzione printf() sul video del PC.

MPLAB - L'AMBIENTE DI SVILUPPO UFFICIALE PER I PICMICRO

Chiunque programmi i PICmicro conosce, o dovrebbe conoscere, il software MPLAB creato da Microchip per integrare in un unico ambiente di sviluppo la miriade di programmatori e compilatori disponibili.

MPLAB è un ambiente di tipo IDE (Integrated Development Environment - Ambiente di Sviluppo Integrato) ovvero un programma unico in cui è possibile editare i sorgenti da compilare, lanciare la compilazione, programmare il PICmicro ed effettuare il debugging, in tempo reale, del programma in esecuzione.
MPLAB può essere scaricato gratuitamente dal sito Microchip al seguente indirizzo:
http://www.microchip.com/1010/pline/tools/picmicro/devenv/mplabi/mplab6/index.htm

In configurazione standard MPLAB include solo il compilatore Assembler MPASM e gestisce solo alcuni programmatori quali il PICSTART ed il PROMATE, ma con l'uso di moduli software denominati plug-in, è in grado di gestire un lunga serie di compilatori e programmatori anche di terze parti.
In questo articolo proveremo l'integrazione con il programmatore ICD2 ed il compilatore C PCH prodotto da CCS.

IL PROGRAMMATORE ICSP ICD-2

Il ICD-2 è un programmatore in-circuit con il quale è possibile trasferire il codice direttamente nella memoria del PICmicro senza doverlo rimuovere dalla scheda in cui viene utilizzato.
L'ICD-2 dispone di una connessione USB verso il PC da cui estrae anche l'alimentazione elettrica. Per chi non dispone di una connessione USB è disponibile in alternativa anche una connessione seriale RS232. In questo caso però occorre collegare l'ICD-2

Figura 1: *L'ambiente di sviluppo MPLAB*

l'alimentatore fornito in dotazione.

Tutta la documentazione ufficiale Microchip sul programmatore ICD2 è disponibile su:
http://www.microchip.com/1010/pline/tools/picmicro/icds/icd2/index.htm

IL REAL TIME DEBUGGING

Oltre alla programmazione, l'ICD 2 consente di effettuare il debug in tempo reale del codice in esecuzione sul PICmicro.
Il debug (dall'inglese togliere gli scarafaggi, ovvero eliminare gli errori di programmazione) è una caratteristica degli ambienti di programmazione che consente l'esecuzione dei programmi una istruzione alla volta per identificare eventuali errori di programmazione.
Fino all'introduzione dei debugger in-circuit, per poter effettuare il debug in tempo reale su un microprocessore occorrevano apparecchiature dal costo molto elevato. Oggi questa situazione per fortuna è cambiata, in quanto le case produttrici di chip hanno inserito nei micro delle circuiterie di supporto al debug che consentono l'esecuzione passo passo di un programma anche con una circuiteria esterna molto semplice. Nel caso di Microchip non tutti i chip dispongono di questa circuiteria per cui si consiglia di verificare sempre se tra le caratteristiche di un chip è presente anche il supporto al debug.
Il PIC18F452 dispone di questa caratteristica.

Figura 2: *Il programmatore/debugger ICD2 della Microchip*

Figura 3: *La scheda PICDEM 2 PLUS usata per l'esperimento in C di questa puntata*

LA DEMO BOARD PICDEM 2 PLUS

Per effettuare gli esperimenti sull'hardware questa settimana abbiamo utilizzato la scheda PICDEM 2 prodotta da Microchip.
Questa scheda dispone di:

- Alloggiamenti per PICmicro a 18, 28 e 40 pin.
- Un display LCD da 16x2 caratteri.
- Una porta RS232.
- 4 led.
- 3 pulsanti.
- Una EEPROM.
- Un beeper piezoelettrico.
- Un connettore per il programmatore ICSP.
- Una sezione millefori da 40x55 mm.

Potete scaricare tutta la documentazione su questa scheda, compreso lo schema elettrico completo, da:
http://www.microchip.com/1010/pline/tools/picmicro/icds/icd2/cupola/pcdmplus/index.htm

PRIMO ESPERIMENTO

Dopo aver installato il software MPLAB, il compilatore PCH ed il software di supporto dell'ICD-2, colleghiamo il programmatore ICD-2 alla scheda PICDEM 2 PLUS. Inseriamo nello zoccolo a 40 pin il PIC18F854, diamo alimentazione alla scheda e proviamo il nostro primo programma in C.

Lanciamo l'ambiente MPLAB dal menu *Programmi > Microchip MPLAB IDE > MPLAB IDE*.
Assicuriamoci che il programmatore di default sia l'ICD-2 come mostrato in figura 4.

Questa è l'icona da premere per attivare la connessione con il programmatore ICD-2.

Nella finestra Output dovrebbe apparire la seguente scritta ad indicare che tutto è andato per il verso giusto:

```
Connecting to MPLAB ICD 2
...Connected
Setting Vdd source to target
Target Device PIC18F452 found,
revision = 0x6
...Reading ICD Product ID
Running ICD Self Test
...Passed
MPLAB ICD 2 Ready
```

Figura 4: *Configurazione dell'ICD-2 come programmatore di default*

```
Connecting to MPLAB ICD 2
...Connected
Setting Vdd source to target
Target Device PIC18F452 found,
revision = 0x6
...Reading ICD Product ID
Running ICD Self Test
...Passed
MPLAB ICD 2 Ready
```

Selezioniamo quindi il menù *Project > New* ed inseriamo nel campo *Project Name,* **esempio1** e nel campo *Project Directory* il nome della directory che vogliamo usare per memorizzare i nostri sorgenti di prova, ad esempio **c:\vitaminac.**

L'MPLAB aprirà una finestra come quella di figura 5 in cui potremo inserire i file appartenenti al nostro progetto.
È consigliabile creare sempre un progetto in quanto l'MPLAB manterrà per noi tutte le informazioni relative alla configurazione del PICmicro, dell'ambiente di sviluppo e dei file da compilare per ottenere il programma finale sotto un unico nome identificativo, il nome del nostro progetto.

Lanciamo a questo punto l'editor ASCII integrato nell'MPLAB selezionando il menu *File > New.*
Sul video apparirà la finestra Untitled. All'interno di questa finestra digitiamo il codice di esempio già utilizzato nella precedente puntata per far lampeggiare i due led:

```
#include <18F452.h>
#use Delay(Clock=10000000)
#fuses HS,NOWDT,NOPROTECT,NOLVP

 void main() {
   for (;;) {
     delay_ms(1000);
     output_high(PIN_B0);
     output_low(PIN_B1);
     delay_ms(1000);
     output_low(PIN_B0);
     output_high(PIN_B1);
   }
 }
```

Salviamolo quindi con il nome esempio1.c nella directory c:\vitaminac dal il menu *File > Save As.*
Aggiungiamo il nostro file al progetto esempio1 selezionando il file sorgente dal menu *Project > Add Files to Project.*

Noteremo che nella finestra di progetto il nostro file sarà stato inserito sotto la voce *Source Files.*

Da ora in poi per accedere al nostro file sorgente sarà possibile effettuare un doppio click sul nome del file nella finestra di progetto.

COMPILAZIONE

Per lanciare la compilazione con il CCS-C assicuriamoci che sia stato selezionato come compilatore di default dal menu *Project > Select Language Toolsuite* selezionando alla voce *Active Toolsuite* il CCS-C Compiler, come mostrato in figura 7.

Premiamo questo pulsante per salvare il progetto.

Premiamo questo pulsante per compilare il programma

Se la compilazione è andata a buon fine dovrebbe apparire nella finestra di output la seguente scritta:

```
Deleting intermediary files...
done.
Executing:
"C:\Programmi\PICC\Ccsc.exe"
"esempio1.c" +FH +DF +LN +T -A
+M +Z +Y=9 +EA
Memory usage:   ROM=0%
RAM=0% - 0%
Loaded C:\vitaminac\esempio1.cof
BUILD SUCCEEDED
```

PROGRAMMAZIONE PICMICRO

Premiamo questo pulsante per trasferire il codice eseguibile nel PICmicro.
Se tutto è andato per il verso giusto avremo nella finestra di output le seguent scritte:

```
Programming Target...
ICDWarn0019: Debug mode is not
enabled. Do you wish to enable
it?
...Erasing Part
...Programming EEPROM Memory
...Programming Program Memory
...Programming Configuration Bits
Verifying...
...Program Memory
...EEPROM
...Verify Succeeded
.. Config Memory
...Programming succeeded
MPLAB ICD 2 Ready
```

Premiamo questo pulsante per lanciare in esecuzione il programma sul PICmicro.

Figura 5: *La finestra progetto dell'MPLAB*

Figura 6: *La finestra di progetto aggiornata*

Se tutto è andato a buon fine dovreste vedere i led RB0 ed RB1 lampeggiare sulla scheda PICDEM 2.

USO DELLE PORTA SERIALE COME CONSOLLE

Nel fare gli esperimenti con il Dev-C++ nelle puntate precedenti abbiamo fatto largo uso della funzione printf() per visualizzare su schermo il valore delle variabili ed i messaggi rivolti all'utente.

Nel caso del PICmicro, non avendo a disposizione uno schermo, dobbiamo utilizzare un'altra soluzione se vogliamo comunicare qualcosa all'utente.

In tutti i compilatori C esiste la possibilità di deviare il flusso di caratteri in uscita generati dalla funzione printf().

Il dispositivo normalmente designato per accettare i caratteri da visualizzare si chiama **stdio** ovvero standard I/O, dispositivo di ingresso/uscita standard. Su PC l'stdio corrisponde per default al video ma qualsiasi altra entità in grado di accettare caratteri potrebbe essere configurata al posto dell'stdio.

Nel CCS-C la funzione printf(), dopo aver formattato correttamente la stringa da inviare all'stdio, richiama la funzione putc() passandogli uno ad uno i caratteri da visualizzare.

Se, ad esempio, chiamiamo la printf() del nostro programma nella seguente forma:

```
  i=100;
  printf("La variabile i vale
%d",i);
```

La printf sostituirà %d con il valore della variabile i, quindi preparerà una stringa con il seguente contenuto finale: "La variabile i vale 100"

A questo punto chiamerà, per ogni carattere contenuto nella stringa finale, la funzione putc() nella forma: putc('L'), putc('a'), putc('v'), e cosi via per ogni carattere fino alla fine della stringa.

Come vedremo tra poco c'é la possibilità di cambiare il codice contenuto nella funzione putc per poter utilizzare i caratteri ricevuti in modo da poterli spedire su un qualsiasi dispositivo d'uscita.

Proveremo ad inviare i caratteri generati dalla printf() verso la porta seriale del PICmicro e quindi al PC.

Il sorgente da utilizzare è il seguente:

```
 #include <18F452.h>
 #use Delay(Clock=10000000)
 #fuses HS,NOWDT,NOPROTECT,NOLVP
 #use RS232(baud=9600,parity=N,
bits=8,xmit=PIN_C6,rcv=PIN_C7)

  void main() {
    int i;

    for (i=0;i<=10;i++) {
      printf("i=%d\r",i);
    }
    delay_ms(10);
  }
```

Questo programma effettua un conteggio fino a 10 ed invia il valore della variabile utilizzata per il conteggio al PC tramite porta seriale.

Per poter visualizzare su PC il testo inviato dal PICmicro possiamo utilizzate il programma HyperTerminal in dotazione con ogni sistema operativo Windows.

Lanciamo HyperTerminal e configuriamolo per lavorare con una delle porte libere del PC.

Configuriamo quindi la velocità di della porta a 9600 baud, no parity, 8 data bit e nessun controllo di flusso. Colleghiamo un cavo non invertente tra la PICDEM2 PLUS ed il PC e compiliamo il programma di prova.

Il risultato che dovremmo ottenere riportato in figura 8.

In pratica con la direttiva:

```
#use RS232(baud=9600,parity=N,
bits=8,xmit=PIN_C6,rcv=PIN_C7)
```

Abbiamo istruito il compilatore CCS ad utilizzare, come standard output, la porta seriale del PICmicro per cui tutte le funzioni che da questo punto in poi presuppongono una uscita verso standard I/O, saranno deviate sulla seriale.

CONCLUSIONI

Vedremo nella prossima puntata come deviare lo standard I/O su display LCD per poi ricominciare con la teoria del linguaggio C.

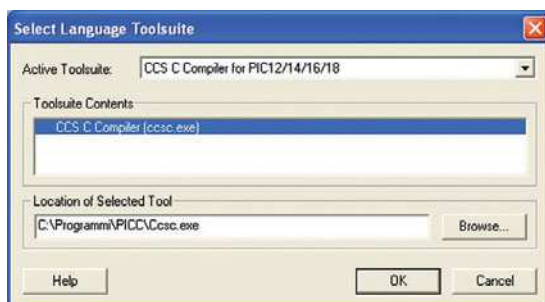



Figura 7: *Selezione del compilatore di default*

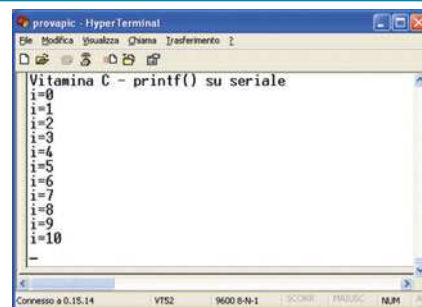



Figura 8: *I messaggi del PICmicro sullo schemo dell'Hyper Terminal*

RICORDO DEI BOLLETTINI TECNICI DELLA GELOSO

di Carlo Pria
c.pria@farelettronica.com

La Società Geloso è scomparsa da più di un quarto di secolo ma il suo ricordo è tuttora ben vivo nella memoria degli appassionati di storia della radio.
Questa azienda è stata una delle pietre miliari dell'industria radiofonica italiana ed i superstiti delle migliaia di apparecchi e milioni di pezzi accessori da essa prodotti, diffusi per tutto il mondo e presenti nelle collezioni di innumerevoli appassionati, stanno a testimoniare la stima che seppe conquistarsi presso un vastissimo pubblico.
Di questa gloriosa azienda ci rimane anche la notissima serie dei suoi bollettini tecnici che vennero distribuiti fino all'ultimo periodo della sua attività e che sono ora ricercatissimi dai collezionisti. Molti di questi appassionati sono anche fra i nostri lettori e da alcuni di essi ci è giunta la richiesta di informazioni riguardo all'introvabile numero 1 di questa collana. Abbiamo per questo pensato di pubblicarlo sulla nostra rivista in modo da consentire a chi ne è sprovvisto di disporre almeno di una ristampa in attesa di trovare l'originale.

PUBBLICAZIONE MENSILE GRATUITA

BOLLETTINO TECNICO S.A.J. GELOSO

N. 1 - FEBBRAIO 1932 - X

Direttore: JOHN GELOSO
Red. Capo: SANDRO NOVELLONE

UFFICI: VIA SEBENICO, 7
MILANO
Tel. 690-288

In questo numero descrizione del ricevitore G. 50

5 Valvole

2 schermate A. F. a "mu variabile„
1 schermata rivelatrice di placca di potenza
1 pentodo finale collegato a resistenza
1 raddrizzatrice
3 circuiti di sintonia accordati a monocomando
attacco per grammofono
uniformità di amplificazione A. F.
altoparlante dinamico.

— PRESENTAZIONE —

Proseguendo il nostro programma, non breve, non agevole, siamo lieti di poter offrire ai dilettanti ed ai costruttori qualche cosa che serva di collegamento tra noi ed i nostri clienti, che sono tutti gli interessati alla radio.

Questo nostro bollettino mensile servirà per noi a far noti i prodotti che con amorevoli cure ogni giorno cerchiamo di rendere più numerosi e perfetti; servirà d'altra parte ai costruttori, ai rivenditori, ai radioriparatori perchè descriverà apparecchi, strumenti, amplificatori studiati nel nostro laboratorio con intendimenti moderni e coi concetti industriali che il nostro direttore tecnico ing. Geloso ha lungamente sperimentato nelle potenti organizzazioni nord-americane.

Affidiamo la redazione del bollettino a Sandro Novellone che colla cooperazione dei tecnici del laboratorio asseconderà nel modo migliore i gusti e le tendenze del momento, fornendo sempre qualche cosa di interessante e di utile.

La Soc. An. JOHN GELOSO

Riprodotto a cura dell'A.I.R.E (1998)

2 BOLLETTINO TECNICO GELOSO

APPARECCHIO A 5 VALVOLE G 50

Uno dei ricevitori più diffusi è il "quattro più una„ realizzato con due alte frequenze e tre condensatori variabili.

Questo apparecchio ha avuto un enorme successo sopratutto nel tipo "midget„ perchè è l'apparecchio medio che può avere le più varie applicazioni, sia come ricevitore economico perchè non costa molto, sia come apparecchio di lusso perchè per i suoi pregi di sensibilità e potenza può essere sistemato in un mobile, anche come radiogrammofono, senza affatto sfigurare.

Per questo ci siamo rivolti a questo notissimo schema desiderando realizzare qualche cosa che interessasse un pò tutti. L'apparecchio è stato studiato nel nostro laboratorio di ricerche durante due mesi. Lo schema definitivo non presenta alcuna novità degna di nota; ma tutto l'insieme, la disposizione dei pezzi ed ogni altro particolare sono stati calcolati partendo dal concetto di ottenere un apparecchio molto semplice, affatto critico, di nessuna difficoltà di costruzione o di messa a punto.

Il disegno del ricevitore ha seguito una evoluzione in seguito alle risultanze sperimentali acquisite sui vari modelli preparati in laboratorio.

Tutti i problemi, ossia quelli riguardanti la alta frequenza la rivelazione e la bassa frequenza sono stati esaminati.

Dalla prima occorre ricavare la massima amplificazione compatibile con la selettività; dalla rivelatrice occorre ottenere la sensibilità necessaria e una tensione di uscita conveniente per il pentodo; dal pentodo occorre avere la massima potenza indistorta.

Diciamo subito che pregio di un buon apparecchio con due alte frequenze sole è quello di arrivare al limite di oscillazione lungo tutta la gamma uniformemente. In questo caso si ottiene la massima sensibilità e selettività senza pregiudicare la qualità della riproduzione, data la presenza di soli tre circuiti accordati. E' insomma necessario ridurre tutti gli accoppiamenti parassitari al minimo, allo scopo di evitare un innesco bloccante ogni ricezione, pur aumentando al massimo il fattore di amplificazione per stadio con l'uso di trasformatori opportunamente studiati e a minima perdita. In seguito si ottiene un grado reattivo più o meno forte diminuendo la schermatura delle bobine al minimo possibile.

Ad uno sperimentatore paziente è poi facile regolare il grado di schermatura in modo che usando qualsiasi tipo di valvola si possa portare il ricevitore nelle migliori condizioni di funzionamento.

Per conseguire le premesse, abbiamo fatto uso di una disposizione molto razionale, che non abbiamo assolutamente più modificato, perchè permette collegamenti cortissimi fra pezzo e pezzo, ossia tra condensatori variabili bobine, valvole e condensatori di blocco.

La schermatura è doppia, ossia interessa le bobine, che sono schermate individualmente e parzialmente, ed il complesso bobina-valvola-condensatore che possiede una schermatura estesa all'intiero stadio.

L'apparecchio si presenta così in modo armonico e semplice: a sinistra l'alimentazione la raddrizzatrice e la valvola di uscita; in centro il condensatore variabile, a destra

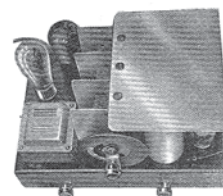



fig. 1 - come si presenta l'apparecchio

lungo il condensatore le tre bobine e le tre valvole di alta frequenza, di cui la prima verso la manopola è quella di aereo e l'ultima è la rivelatrice.

I NOSTRI TECNICI SONO A VOSTRA DISPOSIZIONE

Sotto la base, in centro il blocco integrale dei condensatori fissi divide pure in due porzioni lo spazio; da una parte escono le linguette dei condensatori di filtro per l'a-

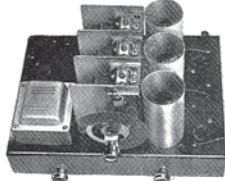



Fig. 2 - l'apparecchio in via di montaggio senza schermo

limentazione, dall'altra, presso le valvole e bobine di alta frequenza stanno gli attacchi dei vari condensatori di blocco relativi a questa parte del circuito.

Gli schermi impiegati sono di 57 mm. con bobina di 25 mm.; questo non è tuttavia un dato rigoroso.

Lo studio delle bobine ha portato al ritorno al vecchio tipo di accoppiamento a trasformatore. Le possibilità del sistema ad impedenza capacità, altrimenti detto a trasformatore impedenza. sono buone sopratutto con l'uso delle valvole tipo 24 e quando è possibile eseguire un lavoro ben fatto, il che non è sempre alla portata di tutti. I risultati ottenuti dal sistema ad impedenza dipendono interamente dalla impedenza stessa.

La costruzione dei trasformatori si vedrà nell'apposito capitolo e nei disegni illustrativi. Teniamo ad aggiungere che i dati costruttivi vanno seguiti alla lettera, per quanto riguarda il primario, perchè dall'accoppiamento induttivo e capacitivo esattamente calcolato, dipende la uniformità di amplificazione lungo la gamma.

Il primario è avvolto direttamente sul secondario e questo comporta un certo accoppiamento capacitivo e la riduzione della capacità dell'avvolgimento primario il quale viene automaticamente a spaziarsi entro l'avvolgimento secondario.

I trasformatori descritti producono una maggiore amplificazione sulle frequenze più elevate.

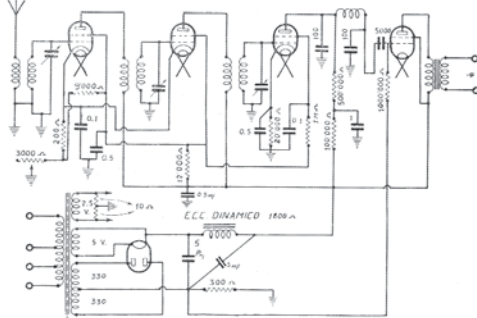



Fig. 3 - lo schema elettrico. Si consiglia aggiungere un condesatore tra rete e terra.

Allo scopo di bilanciare questo fattore, il primario d'aereo è costruito in modo da risuonare su un'onda piuttosto lunga. Esso consiste in una bobinetta di 2400 microhenrys valore ottenibile avvolgendo approssimatamente 300-350 spire come indicato nei dati costruttivi.

Con questo si viene ad ottenere un sistema sufficientemente selettivo. La selettività può essere d'altronde regolata agendo sull'accoppiamento di aereo.

I ritorni delle placche, griglie,-schermo e catodi sono disposti su condensatori risp. di 0,5 0,5 e 0,1 mf. per eliminare gli accoppiamenti. I condensatori non possono scendere sotto il valore indicato, che è il minimo.

La regolazione del volume viene ottenuta

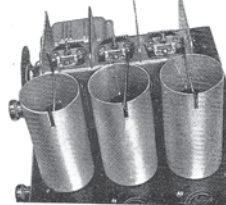



Fig. 4 - Particolare delle bobine e degli attacchi al condensatore; si noti il filo di massa tra bobina schermo e condensatore.

agendo sul catodo. Essa è sicura, efficente e non produce perdite di selettività agendo sulle multimu. Il valore impiegato è di 5000 ohm. Esso non è rigoroso.

Potendo ottenere un reostato a variazione logarimtica, ossia circa 200 ohm di variazione sul primo quarto di giro, 800 nel secondo quarto e poi 9000 l'uso di un reostato di 10.000 ohm sarebbe altamente consigliabile. Disgraziatamente, tali potenziometri sono difficilissimi da trovare in Italia, oppure si hanno a prezzi proibitivi; cosicchè fino a quando Geloso non costruirà un reostato veramente razionale, non sarà possibile impiegare un reostato di valore superiore a 5000 ohm per le difficoltà di manovra che ne seguirebbero.

La valvola rivelatrice è usata per rettificazione di placca. Sul catodo una resistenza di 25.000 ohm. Sulla griglia schermo 1 megohm. Sulla placca 500.000 ohm e un circuito filtro di 100.000 ohm ed 1 mf. Questo filtro abolisce il ronzio e consegue una autoregolazione della tensione anodica. Anche sulla griglia schermo il valore elevato della resistenza ha lo stesso scopo.

I valori indicati servono per ottenere la massima sensibilità. A chi occorresse la massima potenza sulla locale o sulle stazioni più vicine indichiamo i valori di 50.000, 6000 e 300.000 ohm rispettivamente sul catodo griglia schermo e placca. Il valore adottato sul catodo di 25.000 ohm è quello che dà i migliori risultati anche nella riproduzione fonografica.

L'accoppiamento col pentodo è eseguito in modo che passi soltanto la componente di bassa frequenza che viene filtrata attraverso a una impedenza e due condensatori fissi. Col pentodo questo filtraggio ha grandissima importanza per impedire accoppiamenti e sovraccarico da parte della alta frequenza. Il condensatore e la resistenza di griglia sono stati studiati a lungo allo scopo di non introdurre distorsioni. E' bene non superare il megaohm perchè col pentodo si possono avere inconvenienti.

Il filtraggio viene eseguito mediante due condensatori e l'eccitazione del dinamico. Il primo deve essere provato a 1000 volta e la sua capacità non deve essere inferiore ai 4 mf., per eliminare il ronzio dovuto sopratutto alla eccitazione stessa. Un valore di 5 mf. è conveniente. Il secondo condensatore ha una capacità di 3 mf. La polarizzazione di griglia del pentodo è ottenuta per caduta della corrente totale di circa 56 mA., attraverso ad una resistenza di 300 ohm. Il negativo generale che è a

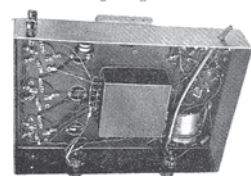



Fig. 5 - L'apparecchio capovolto; Si osservi la disposizione dei collegamenti di alta frequenza, che è assai importante.

massa non è quindi il negativo massimo, al quale viene connesso il ritorno della resistenza di griglia del pentodo. Il centro filamento del pentodo e così pure quello delle altre valvole sono collegati a massa con una resistenza a presa centrale di 20 ohm. Il sistema di polarizzazione è risultato, se non il più razionale, praticamente il migliore.

Il dinamico è del nostro tipo 717 con una eccitazione di 1800 ohm che dà una caduta di circa 100 volta, ed un trasformatore di uscita per pentodo; esso viene collegato mediante un portavalvole, in modo da potersi facilmente distaccare in caso di bisogno.

Sull'apparecchio esiste quindi uno zoccolo di più.

ELENCO DEL MATERIALE OCCORRENTE

1 trasformatore secondo i dati — Geloso N.° 351 —
1 manopola a demoltiplica con finestrella — Geloso —
1 portalampadina per detta — Geloso —
1 resistenza a presa centrale 20 ohm — Geloso CR 20 —
1 resistenza 9000 ohm — Geloso N 9000 —
1 resistenza 12000 ohm — Geloso N 12000 —
1 resistenza 300 ohm — Geloso R300 —
1 resistenza 200 ohm — Geloso R200 —
4 zoccoli UY — Geloso 501 —
2 zoccoli UX — Geloso 503 —
3 bottoni di bachelite — Geloso —
1 altoparlante elettrodin. 1800 ohm — Geloso tipo 717 —
1 blocco condensatori fissi tipo G 50 — Microfarad — con le seguenti capacità:
5 mf. — 1000 volta — 3 mf. — 750 — 1 mf., 0.5 mf. 0.1 mf., 0.1 mf., 0.5 mf., 0.5 mf., — 750 —
Condensatori in mica: 1 5000 cm. 2 100 cm.
1 condensatore carta 0,02, 750 Volta — Microfarad — resistenze 1/2 Watt — Siemens —: 1 500000, 1 100000, 1 25000, 2 1 Megohm
Materiale diverso:
1 blocco condensatori variabili tripli tarati in 5 punti al 0,5 % capacità 375 cm.
1 spina UX per dinamico — ricavabile da 1 valvola —
1 potenziometro o reostato 5000 ohm
1 interruttore a rotazione
Materiale meccanico:
1 chassis ferro 320 × 230 × 60
3 schermi cilindrici 57 × 110 di alluminio 8/10
1 schermo ferro secondo i dati
Minuterie:
materiale per la costruzione delle bobine: cm. 30 tubo

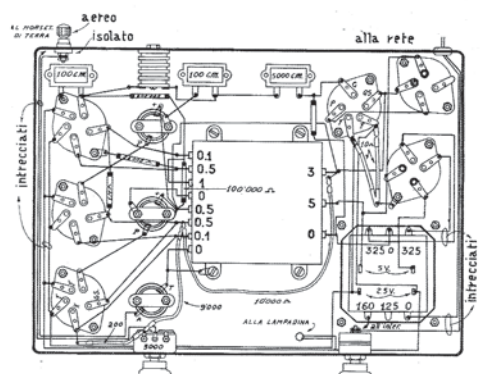



Fig. 6 - Lo schema costruttivo. Il morsetto di terra è nascosto da quello di aereo. Così pure il secondo attacco dell'interruttore. Non è stato segnato il condensatore di 0,02 tra rete e terra. Manca la resistenza di 300 ohm tra centro alta tensione e terra.

bachelite 25 mm. esterno; 35 metri filo 0,25 2 seta 80 metri filo 0,1 2 seta — per primari ed impedenza — 6 viti e squadrette per fissaggio bobine, 12 capofili ed occhielli.
1 rocchetto per impedenza aereo
1 rocchetto per impedenza alta frequenza
35 viti con dado e ranella
1 lampadina 2,5 volta
20 capofili argentati
1 ranella di gomma per passaggio cordone luce
1 cordone e spina luce
1 cordone per dinamico a tre fili
2 morsetti bachelite
2 ranelle isolanti
3 clips per attacco valvole schermate
metri 8 filo per collegamenti.
Valvole: 2 UY 235 — 1 UY 224
1 UY 247 — 1 UX 280

COSTRUZIONE DI PARTI SPECIALI

La costruzione delle bobine è illustrata nelle figure. Queste comportano 150 spire di filo 25/100 2 seta avvolte su di un tubo di bachelite di mm 25 di diametro esterno. Lo spessore del tubo è di 1 mm. La natura e spessore del tubo ha una certa influenza sulle perdite ad alta frequenza. E' bene usare tubo bachelizzato della migliore qualità.

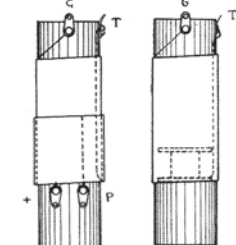



Fig. 7 e 8 - Trasformatori intervalvolari e di aereo. Si noti il senso degli attacchi, e la posizione della bobinetta di aereo.

In caso di dubbio è conveniente paraffinare il tubo per immersione in paraffina caldissima. In questo modo vengono ridotte eventuali perdite dovute alla igroscopicità del materiale.

Le spire secondarie servono per una capacità del condensatore di 375 cm.

Se quindi la capacità massima è differente, le spire vanno aumentate o diminuite fino ad ottenere il piazzamento nella gamma 550-1500 Kc. Le due bobine di accoppiamento tra le valvole sono identiche e comportano un primario avvolto direttamente sul secondario, dalla parte della terra, consistente in 75 spire (o comunque la metà delle spire secondarie) di filo 1/10 2 coperture seta.

Il filo deve essere avvolto direttamente e nello stesso senso. Le spire troveranno posto quindi negli avallamenti tra due spire successive del secondario e quindi le spire primarie rimarranno automaticamente spaziate.

Giunto al termine dell'avvolgimento si fermerà la 75/a spira del primario con una goccia di cera e poi si riporterà il filo al relativo capofilo fissato sulla bobina e che verrà collegato con la placca.

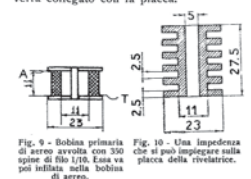



Fig. 9 - Bobina primaria di aereo avvolta con 350 spine di filo 1/10. Essa va poi infilata nella bobina di aereo.

Fig. 10 - Una impedenza che si può impiegare sulla placca della rivelatrice.

L'altro capo o capo inferiore del primario sarà collegato con l'alta tensione.

In totale si troveranno 4 capofili: due in alto della bobina corrispondenti al secondario, e due in basso corrispondenti al primario. Quelli inferiori passeranno attraverso al foro di 30 mm fatto e nello schermo e nella base, sì da essere agevolmente raggiungibili col saldatore.

Ai due attacchi superiori sono collegati l'estremità superiore del secondario, che verrà collegata al cappuccio di griglia della

valvola relativa ed alle lamine fisse del condensatore variabile; e l'estremità inferiore del secondario che, riportata in alto, verrà collegata sia al bordo superiore degli schermi, sia all'attacco delle placche mobili del relativo condensatore variabile. Si tratta, in sostanza, di eseguire un collegamento elettrico diretto del ritorno a terra del secondario al rotore del condensatore variabile eliminando i pericoli di un ritorno a massa non perfetto.

La bobina di aereo possiede un secondario identico alle altre bobine.

Il primario invece è eseguito alla rinfusa su di un rocchetto delle dimensioni segnate in figura, con 350 spire di filo 1/10 2 seta. Avvolgendo la bobina nello stesso senso del secondario l'entrata va all'antenna e l'uscita va alla terra. Comunque il senso degli attacchi qui non ha molta importanza.

Un altro accessorio che può essere eseguito facilmente è l'impedenza. Possedendo una macchina per avvolgere bobine a nido d'api, si può eseguire 1200 spire di 1/10 su un tubo di 11 mm.

In caso contrario si può eseguire un rocchetto come quello di fig. 10 sul quale si avvolgeranno totalmente 1100 spire di filo 1/10 2 seta.

L'impedenza verrà fissata mediante una vite di ottone (non di ferro).

Lo chassis viene costruito con lamiera di ferro dello spessore di 10 o 12 decimi. Le dimensioni sono segnate in fig. 11 Esse possono essere facilmente ridotte a cm. 315×220.
La lastra di ferro, dopo tagliata, viene piegata ed in seguito vengono eseguite delle saldature a stagno sui quattro angoli. Possedendo un buon saldatore, che scaldi sufficentemente la lamiera, questo è il metodo più spiccio. Non è il caso di aggiungere che i fori per il regolatore e l'interruttore possono essere spostati a seconda delle necessità di ordine estetico.

I fori punteggiati servono per i variabili tipo FIMI.

Lo schermo sulle valvole viene costruito

Fig. 11 - Il piano di foratura.

secondo i disegni di fig. 13 in lamiera di ferro od alluminio, e poi applicato con due viti.

MONTAGGIO

Una volta ottenuti tutti i pezzi occorrenti, si può iniziare il lavoro vero e proprio di montaggio. Si incomincerà dagli zoccoli, dai morsetti, regolatore di volume, interruttore; poi si monterà il blocco dei condensatori fissi, l'impedenza e le bobine ad alta frequenza, poi il condensatore variabile e la manopola, da ultimo il trasformatore di alimentazione.

Con alcuni tipi di condensatori tripli occorre invece montare prima quelli e poi il blocco dei condensatori fissi.

Prima di fissare il condensatore triplo si provvederà il mezzo di fissare su di esso mediante due viti lo schermo per le valvole e bobine ad alta frequenza.

Fig. 12 - Particolare degli attacchi tra bobina, valvola e condensatore.

Si inizieranno i collegamenti da quelli di accensione che vanno intrecciati e fatti con conduttore a grossa sezione. Di mano in mano si faranno gli altri collegamenti seguendo fedelmente lo schema costruttivo almeno per tutto quanto riguarda la alta frequenza.

I collegamenti di alimentazione dei catodi, delle griglie schermo e dei ritorni di placca delle alte frequenze vanno fatti sempre arrivando direttamente al blocco condensatori. Non bisogna unire insieme, per esempio, i due catodi e poi collegarne uno al condensatore di blocco, bensì entrambi i catodi separatamente verranno collegati al condensatore.

Un filo che occorre schermare è quello tra il serrafilo di aereo e la bobina di aereo. Per fare questo, si intreccia questo collegamento con un'altro filo morto, che si pone a terra da una parte e dall'altra; cioè da una parte al serrafilo di terra (che è massa) e dall'altra col cursore del reostato, pure a massa.

Intrecciati vanno anche i due fili tra i 330 Volta del secondario alta tensione e le due placche.

Nella parte superiore dello chassis si devono solo eseguire gli attacchi (intrecciati) al portalampadina e quelli tra i ritorni a terra delle bobine e gli schermi cilindrici e tra questi ed i rispettivi rotori del condensatore triplo. Inoltre gli attacchi di griglia delle bobine, che pure sporgono superiormente, si collegheranno ai clips delle valvole da una parte, e dall'altra alle lamine fisse (statori del condensatore triplo). Con questo termina il montaggio.

Il filo impiegato può essere l'8/10 stagnato ricoperto con tubetto sterling. Per i collegamenti di accensione esso può essere messo in doppio.

Restano da eseguire i collegamenti alla rete ed al dinamico. Per il primo si impiegherà un cordone normale a due fili con con spina. Il foro di passaggio del cordone attraverso lo chassis può per sicurezza essere guarnito con una apposita rosetta di gomma.

Il cordone del dinamico porta tre fili i quali vengono saldati ad una spina a 4 piedini. Non possedendo una spina, si può ricavarla da uno zoccolo di una vecchia valvola UX. Il filo proveniente dalla placca verrà connesso al primario del trasformatore del dinamico (capofilo marcato P). Il

secondo capo del primario, marcato +B. si collegherà con uno dei capi della eccitazione. Questi sono i due capofili a destra vicini uno all'altro e segnati "1800 Ω".

Lo stesso capofilo + B si collegherà attraverso al cordone al positivo generale dell'apparecchio, ossia al condensatore di 3 mf. L'altro capofilo dell'eccitazione si collegherà al positivo massimo, ossia al capo del filamento della 280 che è connesso col condensatore di 5 mf.

Avvertiamo che nel costruttivo manca la resistenza di 300 ohm collegata tra il centro alta tensione del trasformatore (325-0-325) e la massa; così anche il condensatore, di 0,02 mf., che riteniamo molto utile, tra un capo della rete e la massa.

Fig. 13 - Lo schermo della parte in alta frequenza. Materiale: alluminio o ferro 1 mm.

COLLAUDO

Assicuratisi che tutto è a posto si inseriscono le valvole e si eseguiscono le misure di tensione come da tabella allegata. Le valvole in alta frequenza possono essere anche le 51 sebbene l'apparecchio sia stato previsto per le 35. Collegato un piccolo aereo al morsetto di antenna (la terra può anche non essere collegata) si deve sentire, col potenziometro al massimo, il rumore di fondo che indica la sensibilità dell'apparecchio. Si prende una stazione piuttosto forte e si regolano i compensatori sino ad avere la massima forza. La stessa cosa si può fare per più stazioni deboli per ultimare la messa a punto; una sulle onde corte una sulle medie ed una sulle lunghe. Se l'apparecchio ha poca tendenza ad oscillare su qualche gamma di onde, si curerà l'allineamento perfetto in questa parte di gamma, trascurando magari le altre, in modo da bilanciare l'effetto.

La tendenza eccessiva alla oscillazione può provenire da diverse cause: cordone del dinamico troppo vicino alla rivelatrice (raro) filo di antenna troppo vicino alla rivelatrice; mancanza di terra (raro) antenna troppo corta; Schermaggio insufficiente; collegamenti malfatti; valvole non adatte o difettose.

Così come ora è descritto lo schermo, serve per valvole "normali" Con alcune marche di valvole è stato necessario curare di più la schermatura.

Spesso si hanno fenomeni di instabilità, sopratutto se la messa a terra dei vari attacchi non è perfetta. Un piccolo strato di vernice sotto una vite o sotto gli schermi cilindrici può pregiudicare il funzionamento dell'apparecchio. A questo proposito conviene collegare con filo di rame, saldato, tutte le viti che collegano a massa i vari circuiti.

Anche il condensatore di 100 cm. sulla placca della rivelatrice può essere aumentato a 200 o 300 se questo si mostrasse necessario.

Comunque, se l'apparecchio è troppo stabile esso è pure poco sensibile. In questo caso sono i compensatori male regolati, oppure bobine mal costruite, o tensioni insufficienti.

La selettività può essere regolata agendo sull'accoppiamento del primario di aereo, ossia spostando, ove occorresse, la bobinetta primaria. Abbassandola la selettività aumenta; occorre però ritoccare il primo compensatore per ogni spostamento. La posizione data, nelle condizioni delle prove, rappresenta un buon compromesso.

Sullo schema non è segnato un condensatore che è spesso utile sia per il ronzio di modulazione sia per la stabilità dell'apparecchio. Esso è posto tra un capo della rete e la massa. Il valore di questo condensatore è 0,02.

Per quanto riguarda il ronzio, esso dipende in maggiore grado dal 1.° condensatore. Questo non può essere ridotto sotto i 4 mf.

Con i valori indicati il filtraggio è più che sufficiente.

Usando almeno 1 elettrolitico di 8 mf. (come primo al posto di quello di 5 mf.) il livellamento diviene assolutamente perfetto. Ciò sarà possibile e comodo fare quando avremo gli elettrolitici. Allora i due condensatori di 5 e 3 possono essere sostituiti con grande vantaggio dagli elettrolitici.

L'attacco del "pick-up" deve essere eseguito sulla griglia della rivelatrice e perciò occorre staccare il clips e collegare il cappelletto della rivelatrice ad un capo del "pich-up" mandando l'altro a terra.

Volendo una installazione stabile si può tagliare il ritorno a terra della bobina della rivelatrice inserendo in serie il diaframma mediante un "jack" che ponga in corto circuito a massa il detto ritorno quando il "pick-up" viene staccato. Occorre prestare attenzione a che i collegamenti del "pick-up" non producano accoppiamenti in alta frequenza. L'amplificazione ottenuta non è molto forte, ma più che sufficiente per usi domestici. In ogni caso occorre adoperare un "pick-up" piuttosto forte a alta impedenza.

Il funzionamento di tutto il complesso, una volta ultimata la semplicissima messa a punto, è molto buono; in località normali la ricezione di una trentina di stazioni con 3 o 4 metri di antenna deve essere possibile. In città i disturbi parassiti danno in alcuni casi molta noia, sopratutto usando la presa di terra come antenna, oppure la rete luce, e perciò questo non è consigliabile. In diversi casi i parassiti sono di origine locale e possono essere eliminati; consigliabile in proposito la lettura del manuale del Mottola (*) noto specialista in materia.

In conclusione crediamo di poter raccomandare questo nostro apparecchio, perchè costa poco e va bene. La migliore pubblicità per noi è assistere i nostri clienti; così come li agevoliamo colla descrizione di un apparecchio studiato nei laboratori "Geloso", siamo a disposizione per qualsiasi altro schiarimento, consulenza, aiuto necessario.

L'unica nostra ambizione è di poter avere, presto, tutto il materiale necessario, tutti i componenti. La strada è lunga, ma ogni giorno un nuovo gioiello viene ad arricchire la collana. Geloso si afferma.

SANDRO NOVELLONE

(*) Dott. Ignazio Mottola - disturbi alle ricezioni radiofoniche - presso Ufficio Stampa Radio, Via Montenapoleone, 1

G 50 – TABELLA DELLE TENSIONI

NB. – Le tensioni sono misurate direttamente tra i piedini delle valvole e la massa. Lo strumento ha una resistenza di 1000 ohm per volta. Le tensioni sono quelle misurate, e non le effettive. Un' approssimazione del ± 10 % è spesso tollerabile e rientra nelle comuni variazioni della rete.

VALVOLE	K (catodo o filamento)	G. S. (Griglia schermo)	P (Placca)
1.a – 551	3 *	90	220
2.a – 551	3	90	220
3.a – 224	1,2	25	65
4.a – 247	— 17 (2)	220	212
5.a – 280	325	---	330 (ALTERNATA)

(*) Le misure sono effettuate col regolatore di volume al massimo.
(2) Tra massa e ritorno di griglia

CORRENTE TOTALE 55-60 mA.

— NEL PROSSIMO NUMERO —

Descriveremo un amplificatore particolarmente studiato per i costruttori, di cui forniremo tutto il materiale occorrente sotto forma di scatola di montaggio. 5 Watt di potenza d'uscita utile assicurano un conveniente uso nei piccoli impianti per film sonoro o locali pubblici. Servono fino a 200 posti di cinematografo o trenta coppie di ballerini. Non nascondiamo l'opportunità di spingere questo ramo affine alla radio sopratutto durante la stagione estiva. Un impianto di riproduzione dei suoni costa poco e costituisce un facile guadagno. Inoltre non è soggetto a particolari licenze Governative.

L'amplificatore possiede 2 valvole tipo 227 come amplificatrici. Lo stadio finale è costituito da due '45 in push – pull. L'alimentazione è fatta con una 280 che eccita anche un dinamico. Forniremo anche lo schema di un apparecchio a due valvole, con dinamico, per la locale.

Questo apparecchio ora in prova dà risultati molto buoni per quanto riguarda la potenza e la purezza del suono. Inoltre esso può servire come amplificatore grammofonico.

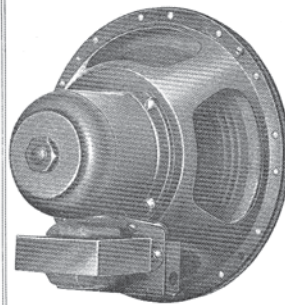
Prossimamente daremo tutto un piano ad uso dei radioriparatori, per estendere la loro proficua attività interessando una sempre più larga clientela. Oggi la professione del radioriparatore può costituire una notevole sorgente di profitti.

Per il numero della fiera campionaria descriveremo un ricevitore tipo supereterodina con tutti i più recenti perfezionamenti della tecnica.

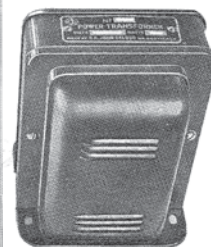
Questo apparecchio è già in istudio nel nostro laboratorio di ricerche.

ESSI SONO RICCHI DI DATI TECNICI

BOLLETTINO TECNICO GELOSO

Questo è il nostro dinamico.
Esso racchiude mesi di esperienze e di studi.
La qualità della carta, la rigidezza del sistema centrante, le caratteristiche elettriche formano un tutto armonico perfetto.
L' Altoparlante Geloso conserva gelosamente la chiara limpidità della musica, il timbro ed il carattere di ogni voce.
Un'apparecchio che possiede un dinamico Geloso si distingue tra mille.

ULTIMA NOVITÀ GELOSO

la manopola a demoltiplica nel tipo normale ed a visuale intera. Frizione fortissima e dolce: mascherine di nuovo stile costituiranno il successo di queste manopole.

QUESTO È IL TRASFORMATORE Tipo 351 USATO NELL'APPARECCHIO G. 50

RICHIEDETE I NOSTRI LISTINI

N. 101 - Trasform. di bassa frequenza
N. 105 - Zoccoli per valvola
N. 108 - Trasform. di alimentazione
N. 110 - Resistenze flessibili
N. 112 - Resistenze con presa centrale
N. 114 - Trasf. alimentaz. serie 301
N. 116 - Altoparlanti elettrodinamici
N. 118 - Manopole a demoltiplica

S. A. JOHN GELOSO - Milano
VIA SEBENICO, 7 - TEL. 690-288

UFFICIO COMMERCIALE:
F. M. VIOTTI - C. Italia, 1 - Tel. 82126

Direttore Responsabile A. NOVELLONE

SPIRIT, UNA MISSIONE, UN ROBOT

prima parte

di Riccardo Ricci
r.ricci@farelettronica.com

Milioni di dollari di investimento, un obiettivo che potrebbe condizionare il futuro dell'umanità, una sola possibilità, un team, un robot: inizia l'avventura su Marte.

Il rover della missione Spirit rischia di diventare il più famoso robot esploratore della storia. Dopo un viaggio di circa sette mesi è arrivato su Marte portando con sé una strumentazione che dovrebbe dare una risposta indiretta alla domanda più entusiasmante e importante per il genere umano. Il veicolo-robot è infatti attrezzato per cercare di confermare che in passato sul pianeta erano presenti attività di acqua in forma liquida. Questa risposta alimenterebbe le speranze di trovare anche tracce di forme di vita basandosi sul fatto che sulla Terra ovunque c'è acqua c'è vita, anche in condizioni estreme e apparentemente impossibili.

Il rover è solo ed è in un ambiente ostile: deve provvedere al proprio sostentamento energetico, deve orientarsi e osservando ciò che lo circonda deve raggiungere i propri obiettivi di ricerca muovendosi tra le polveri e le rocce scegliendo il proprio percorso fino a quando il rapido degenerarsi delle batterie ricaricabili lo trasformerà solo in un immobile ricordo artificiale del panorama marziano. Una appassionante avventura, una sfida che è solo la punta dell'iceberg di un sistema estremamente complesso come solo una missione spaziale può esserlo, un progetto nel quale ogni più piccolo e insignificante dettaglio può distruggere in un istante le speranze della scienza e migliaia di dure ore di lavoro.

Figura 1: *Il rover, il veicolo-robot a cui è affidato l'importante compito di ricerca*

LA MISSIONE

La distanza tra la Terra e Marte varia continuamente a causa del movimento che pianeti compiono lungo le loro orbite. Ogni due anni circa, i due corpi si trovano però in un punto nel quale la loro distanza è la minima raggiungibile ed è in questo periodo che si aprono le "finestre di lancio", cioè dei momenti nei quali è possibile inviare missioni verso Marte con il minimo dispendio di energia e tempo di viaggio.
Ciascuna missione viene sempre pianificata e programmata in anticipo in funzione delle finestre utili per il lancio; non riuscire a partire nei giorni previsti (con tolleranze di qualche settimana) significherebbe dover rimandare la missione di 2 anni e trovare nuovi fondi per il mantenimento delle relative strutture.
Avendo una data di partenza non modificabile e non rimandabile è quindi necessario non fallire e farsi trovare pronti all'appuntamento.
Questo significa avere collaudato il funzionamento e l'affidabilità di ogni componente e di ogni aspetto della missione ma potendo contare sugli immancabili imprevisti è sempre necessario disporre anche di una certa dose di fortuna. Queste stesse parole sono state pronunciate da Steve Squyres, uno dei responsabili scientifici del rover, e lasciano trasparire quanto siano sottili e spesso impossibili da prevedere le situazioni che si devono e che si dovranno affrontare, anche al di là del sempre possibile errore umano. Valutando a grandi linee i numeri e le situazioni da affrontare possiamo brevemente renderci conto delle diverse difficoltà operative.
Come prima cosa è necessario trovare fondi per milioni di dollari per

finanziare la missione, quindi si deve scegliere un veicolo che con cui portare il carico al di fuori della nostra atmosfera, capace cioè di alzarsi in aria e spingere tutto il suo peso a velocità elevatissime. E per quanto la tecnologia sia collaudata, ogni lancio è una sfida a sé: si ha a che fare, ad esempio, con enormi quantitativi di propellenti solidi e liquidi (ossigeno e idrogeno mantenuti a temperature di –183 e –250 gradi centigradi) che già da soli rappresentano una situazione estrema dipendente da tantissime altre variabili (funzionamento dei computer, rilevazioni dei sensori, condizioni di sicurezza, ecc). gestite e controllate sia automaticamente che dal centro di controllo.

Figura 2: *Gli stadi del razzo che hanno portato il rover su Marte*

Una volta lasciata la rampa di lancio è necessario orientarsi correttamente e seguire la traiettoria prevista, spingersi ad oltre 22mila Km/h, separare i vari stadi (booster) al momento opportuno e lanciare nello spazio il veicolo che compirà il viaggio verso Marte, tutto in qualche decina di minuti.

Il veicolo spaziale dovrà quindi orientarsi calcolando la propria posizione autonomamente, basandosi sulla posizione delle stelle e con il riferimento del nostro Sole.
Una passeggiata di oltre 200 giorni compiuta sul filo dei millimetri, nella quale ogni eventuale variazione o correzione di traiettoria comporterebbe la perdita di un po' di quella indispensabile e preziosa energia necessaria per il viaggio. Anche in questa favola, il lupo è sempre in agguato e oltre alle temperature estreme (negative) dello spazio si devono considerare tanti altri fattori capaci di compromettere la passeggiata nel bosco. È possibile ad esempio essere colpiti o colpire frammenti e polveri vaganti che a causa della alte velocità in gioco perforerebbero il veicolo, da parte a parte, come micidiali proiettili. Anche se potessero essere rilevati in anticipo, ogni manovra anticollisione sarebbe probabilmente così dispendiosa da compromettere comunque il lungo viaggio. Ancora, ricordiamo che i sistemi di navigazione e i computer di bordo sono componenti elettronici che per quanto affidabili lavorano entro certe tolleranze di temperatura mantenute automaticamente all'interno del veicolo. Ma il loro funzionamento può essere compromesso da radiazioni o tempeste solari come è già capitato in passato e in questi casi c'è solo da sperare che il sistema sia in grado di sopportare e riprendere correttamente il proprio lavoro. Forse ricorderete che lo scorso anno molti satelliti per telecomunicazioni sono stati messi temporaneamente fuori uso da una tempesta solare che ha compromesso, per un certo periodo, le comunicazioni terrestri con i telefoni cellulari GSM. E anche sulla Terra, lo stesso fenomeno può distruggere letteralmente trasformatori e linee elettriche.

Ammesso quindi che il viaggio proceda senza problemi, si raggiunge l'obiettivo e ci si scontra con una nuova serie di fasi molto delicate: Spirit dovrà atterrare sulla superficie di Marte in una zona già definita lungo l'equatore del pianeta, zona che oltre ad essere la più adatta per condurre gli esperimenti e le rilevazioni scopo della missione, garantisce anche la quantità di luce solare necessaria per ricavare energia sufficiente al sostentamento del robot.
È quindi necessario sapere esattamente dove entrare nell'atmosfera marziana calcolando anche l'angolo di impatto con la massima precisione per evitare di rimbalzare su di essa (e perdersi nello spazio) o di produrre attriti troppo elevati, con conseguente sviluppo di temperature impossibili da sopportare.
(Proprio la temperatura ha compromesso il rientro dell'ultima missione Shuttle, disintegratasi a causa del cedimento della protezione termica del veicolo, danneggiata in partenza).

Superata questa fase, inizia la discesa nell'atmosfera del pianeta e nonostante gli attriti calcolati, la velocità del veicolo è di circa 470 Km/h, ancora troppo elevata per un consentire un atterraggio morbido... La

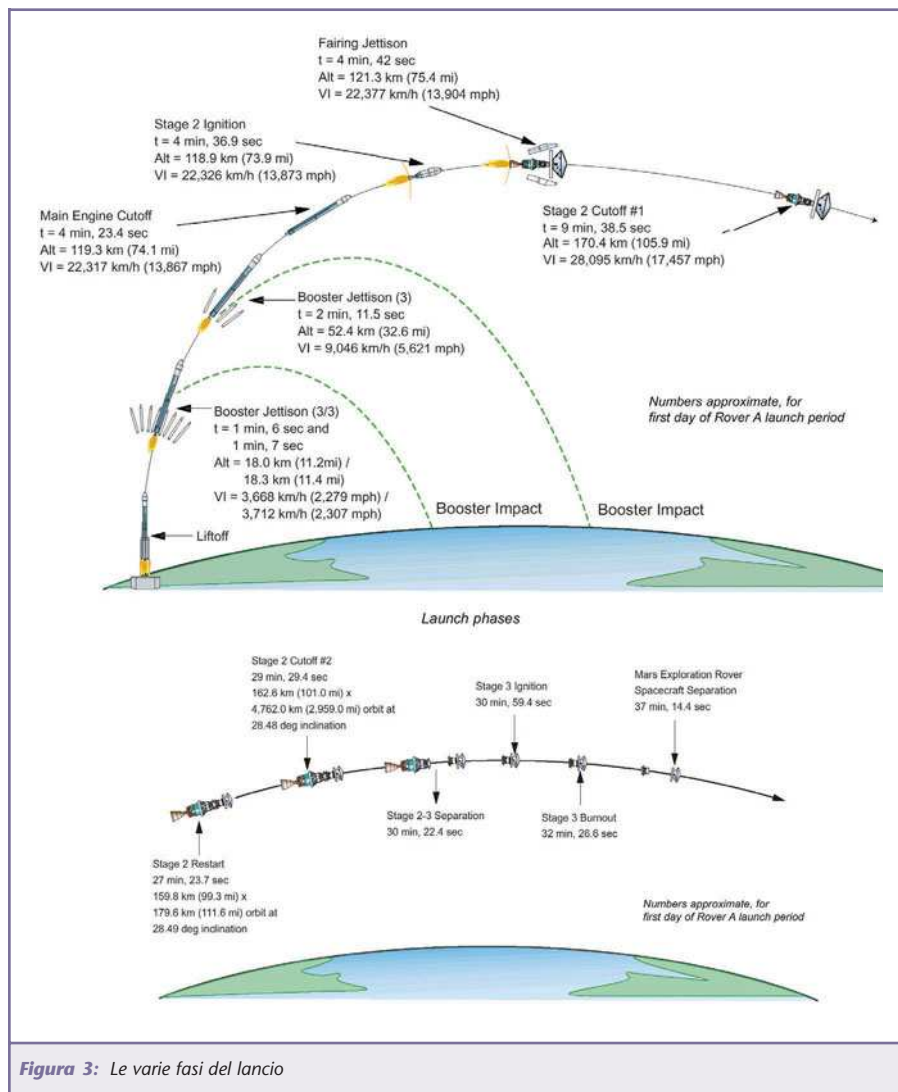



Figura 3: *Le varie fasi del lancio*

soluzione è quindi antica e collaudata come un paracadute, che deve aprirsi esattamente a 8,6 Km di altezza dal suolo (mancano circa 100 secondi all'atterraggio).
Monitorando la propria posizione e altezza via radar, il sistema deve fare uscire il rover in tempo utile, (vedremo tra poco in che modo), azionare dei retrorazzi frenanti e quindi sganciare il suo carico a qualche decina di metri dal suolo, ad un'ultima

Figura 4: *La grande galleria del vento nella quale si sono compiuti i test sul paracadute, decisivi per il corretto funzionamento nell'atmosfera di Marte*

velocità rilevata di circa 80 Km/h.
Ancora in volo, il robot viene quindi sganciato dal veicolo principale e rimane all'interno del proprio involucro dal quale fuoriescono una serie di grandi air-bag che serviranno ad attutire l'impatto che avviene, a questo punto, in caduta libera. In soli 10 minuti dall'ingresso nell'atmosfera marziana, gli airbag toccano terra producendo una lunga serie di rimbalzi. Solo dopo 66 minuti gli airbag si sgonfieranno per liberare l'involucro protettivo che avvolge il rover.
L'atterraggio è quindi una fase nella quale possono verificarsi molti eventi compromettenti. Per questa missione, si è cercato di puntare su elementi già collaudati in missioni precedenti quali la capsula di ingresso nell'atmosfera, l'uso del paracadute e l'impiego degli airbag. In questo caso però, il peso complessivo del robot esploratore era superiore a quello dei suoi simili e tutto il sistema è stato riconsiderato in proporzione, tranne che per le dimensioni. Come prima cosa, si è realizzato un paracadute più grande e sono state compiute delle prove nella nostra atmosfera aumentando in proporzione il peso del veicolo per simulare lo "strappo" e l'apertura alla stessa velocità di discesa raggiunta su Marte. L'ottimismo dei progettisti si è però scontrato con i risultati del primo test, che ha rovinosamente squarciato il paracadute in più punti facendo precipitare il peso che simulava il rover. Si è quindi dovuto ripensare il paracadute in un periodo di tempo già molto vicino al lancio e la nuova soluzione è stata approntata solo qualche settimana prima. Il nuovo test è stato compiuto con una simulazione in una gigantesca galleria del vento della Nasa, capace di sparare aria in uno spazio così grande da permettere la totale apertura del paracadute e simulare la reale velocità che avrebbe dovuto affrontare. Anche questo test è però fallito, questa volta non strutturalmente (il paracadute ha resistito allo strappo) ma funzionalmente: il paracadute si è aperto ma la sua dimensione in rapporto alla velocità dell'aria non gli permetteva di aprirsi correttamente: rimaneva riversato su di sé e senza fornire la giusta resistenza necessaria per ridurre la velocità al valore previsto.

Se tale situazione si fosse riproposta su Marte, Spirit sarebbe caduto al suolo con conseguenze non immaginabili... Era necessario correre ai ripari, trovare nel più breve tempo possibile tutti gli esperti del settore e risolvere il problema. Milioni di dollari in tecnologia stavano per essere

compromessi da uno strumento fatto di tela e corda e la tensione dei progettisti era al massimo, così come la pressione e la consapevolezza di avere sulle spalle tutto il peso della missione. La soluzione arrivò però in tempo e si concretizzò con la riduzione del foro di sfiato del paracadute, grazie all'intervento di esperti che già conoscevano il fenomeno della mancata corretta apertura. Utilizzando proprio ago e filo, il paracadute venne modificato e provato con esito positivo durante i successivi test.

Un test simile venne effettuato in grande tranquillità anche con gli airbag, strumento sicuro ed affidabile già sperimentato più volte. A compromettere la funzionalità del sistema ci sono però le incognite e le variabili del punto di atterraggio, che è sicuramente formato da rocce di cui non è però possibile prevedere la forma né la disposizione.
Se l'airbag cadesse su una roccia appuntita alta due metri, gli airbag verrebbero facilmente infilzati e resi inefficaci. Ma i tecnici erano ottimisti e il test simulato in situazioni regolari venne considerato solo una formalità. Un formalità che però, fallì tra lo stupore generale. La visione rallentata del filmato con la caduta da pochi metri su un terreno inclinato mediamente roccioso mostrava l'aprirsi di uno squarcio sull'airbag proprio al primo impatto. Un allarme, anch'esso fatto di tela, filo e aria, che rischiava di vanificare nuovamente la missione. La soluzione arrivò anche in questo caso con estrema rapidità e con camere d'aria indipendenti e rinforzate che superarono i successivi test senza alcun cedimento, con grande sollievo da parte di tutti.

Figura 5: Gli airbag utilizzati per ammorbidire l'atterraggio del rover, in un test compiuto sulla Terra

Nel frattempo i primi test compiuti sulla resistenza delle delicate apparecchiature del laboratorio robot erano state condotte con successo, così come la funzionalità dello stesso rover e delle sue parti.

La missione era finalmente pronta e la data di partenza era ormai prossima. Tutti i problemi riscontrati erano stati affrontati e risolti, il vettore era già sulla rampa di lancio e il conto alla rovescia veniva seguito a distanza di sicurezza dai responsabili che per mesi avevano lavorato senza sosta.

"Meno 10, 9, 8, 7,... sospendere! sospendere!".
L'avventura si fermò a –7 secondi dal lancio. Venne improvvisamente rilevata una anomalia di funzionamento nell'erogazione del combustibile. Il razzo e tutto il suo contorno erano appena diventati una potentissima bomba incontrollabile. Incredibile! Anni di lavoro, e per soli 7 secondi tutto poteva saltare in aria distruggendo idee, speranze, scienza, tecnologia, risposte e il morale dei progettisti, questa volta impotenti dinnanzi ad una situazione che solo altre competenze avrebbero potuto gestire.
Il malfunzionamento venne individuato, analizzato, riparato e la potente bomba incontrollabile tornò ad essere un messaggero di speranza pronto ad un nuovo count-down che, questa volta, giunse al termine.
Il 4 Gennaio 2004, la missione Spirit portava il rover sul suolo di Marte.

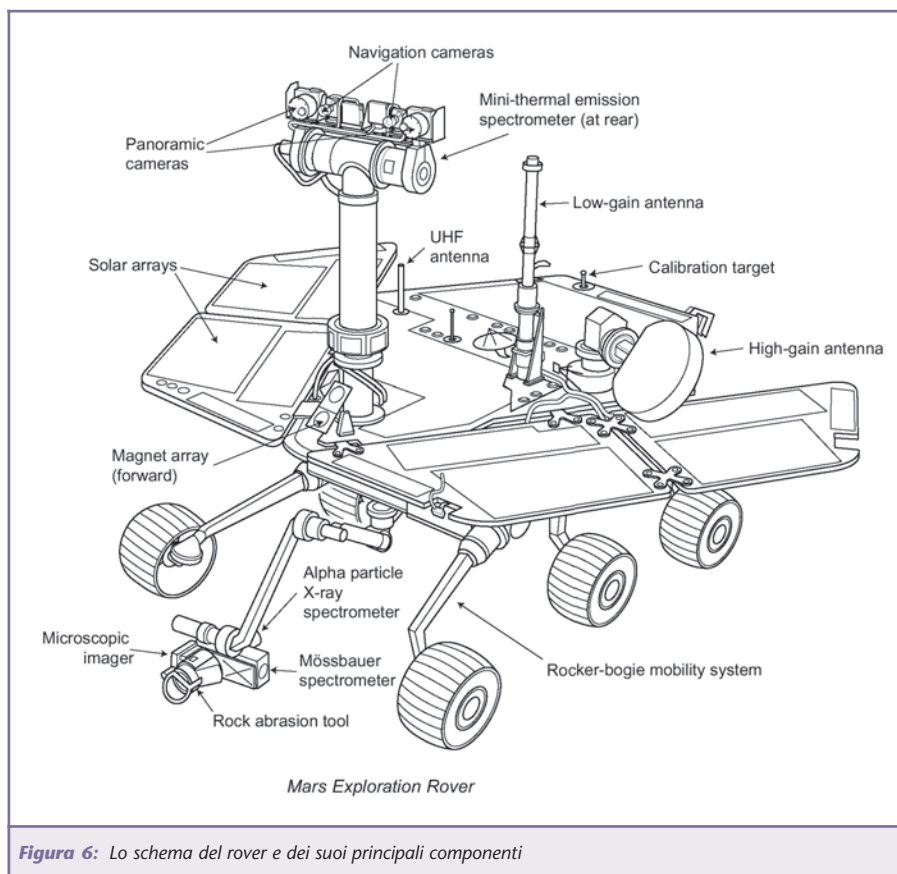



Figura 6: Lo schema del rover e dei suoi principali componenti

Tutte le fasi successive al lancio erano state affrontate con successo. L'esplorazione al suolo poteva avere inizio.

IL ROVER

Il rover è lo strumento attraverso il quale si concretizza la missione sulla superficie di Marte. È un piccolo veicolo robot a 6 ruote, dotato di strumenti e apparecchiature da laboratorio che gli permettono di compiere esperimenti e rilevazioni sul terreno e sulle rocce del pianeta.
Il rover è composto da un insieme di tecnologie elettroniche e meccaniche assemblate intorno ad una struttura che protegge i suoi organi vitali. Le sue parti esterne principali sono composte da un braccio meccanico attrezzato, da una sorta di "testa" dotata di camere per la ripresa, da parti mobili disposte su gambe che sostengono le ruote, da pannelli solari e da strumenti e antenne per la comunicazione.
La struttura portante è formata da una scatola, il WEB (acronimo di warm electronics box) protetta da diversi strati che isolano i computer e le batterie situati in un ambiente termocontrollato.
(Durante la notte, la temperatura del pianeta arriva a circa -96°C). Sulla parte superiore del WEB è alloggiata una superficie triangolare che ospita i pannelli solari e che può essere attrezzata.

È importante sottolineare che i progettisti di questo rover, oltre a renderlo efficace per le funzioni a cui esso è destinato, hanno dovuto escogitare un sistema di "trasformazione" molto simile a quello attuato dai robot dei cartoni animati giapponesi. Armi atomiche a parte, il problema reale nasceva dalla necessità di utilizzare per il viaggio un veicolo la cui capacità di carico era stata studiata per il precedente modello di rover. Un po' per economia (i budget per ogni missione non sono più alti come un tempo) e un po' per il positivo collaudo già effettuato, l'impegno dei progettisti si è quindi dovuto rivolgere anche verso la capacità di adattare un grande rover nello spazio di un piccolo rover... Ed ecco spuntare articolazioni, movimenti e contorsionismi capaci di raccogliersi su loro stessi, un po' come faceva Alien quando si annidava appeso nei corridoi bui della malcapitata astronave. Molto più simpatico e amichevole, il rover è un bel prodotto dell'inventiva umana che la tecnologia ha reso concepibile a dispetto dei rigidi limiti di spazio imposti.

Figura 7: Il WEB contiene all'interno tutti gli organi vitali del robot

IL COMPUTER DI BORDO

Il cervello del rover è chiamato REM (Rover Electronics Module) ed è composto da un computer le cui caratteristiche sono comparabili a quelle di un normale computer portatile, dotato però di una speciale memoria resistente alle radiazioni e in grado di mantenere i propri dati anche in caso di black-out. Le specifiche del computer non sono affatto esagerate e sia la potenza di elaborazione che la quantità di memoria (128 MB di DRAM e 3MB di EEPROM) sembrano essere molto modeste se paragonate alla complessità delle funzioni richieste durante la missione. Tuttavia, è certo che tale computer viene sfruttato al 100% e in modo totalmente diverso da quanto siamo abituati a vedere; questo fa parte di un accurato progetto di ottimizzazione che non richiede niente di più di quanto viene realmente utilizzato. (A titolo di paragone, la precedente missione Pathfinder disponeva di una quantità di memoria 1000 volte inferiore !)
Il computer è interfacciato con l'esterno per mezzo di un VME bus (Versa Module Europa), uno standard industriale che permette lo scambio di dati con i motori, gli strumenti scientifici di bordo e con le funzioni di comunicazione.

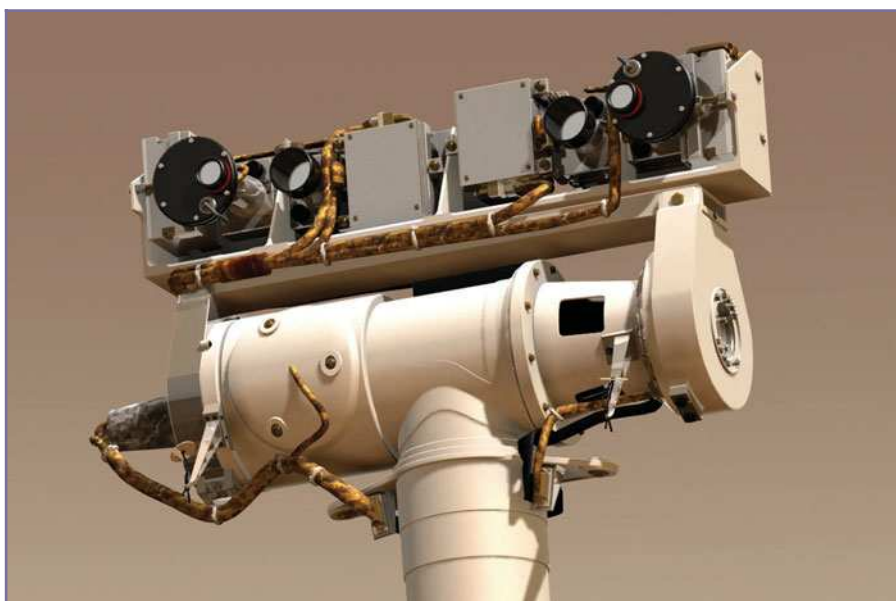



Figura 8: Particolare della testa: alle estremità, coperte dai dischi neri portafiltri, le due PanCam; più all'interno le Navicam

La cosa che emerge da queste informazioni è il fatto che la tecnologia informatica utilizzata sul rover pare sia alla portata di chiunque e che molti di noi possiedono o usano già tecnologie simili per svolgere compiti sicuramente molto diversi e quasi sempre meno complessi.

I SENSORI PRINCIPALI

Il rover è in grado di riconoscere la propria posizione sul piano grazie un sensore inerziale a 3 assi (IMU) che restituisce l'angolo di inclinazione della struttura sui tre piani dello spazio. Altri sensori si occupano di registrare informazioni che mantengono cosciente il rover della proprio "stato di salute", rilevando ad esempio la temperatura, fattore della massima importanza per il corretto funzionamento di tutto il sistema.

Un altro importante dispositivo di senso è formato da quello che chiamano "neck and head" (collo e testa) cioè una sorta di albero sul quale poggia un sistema di camere. Grazie al lungo collo, la visione dell'ambiente circostante avviene da circa un metro e mezzo di altezza e offre un punto di prospettiva quasi umano. L'acquisizione delle immagini è affidata ad una Pancam (Panoramic Camera) che utilizza due sensori CCD affiancati (a colori e ad alta risoluzione) per fornire immagini "stereo", cioè dotate di profondità. Il collo sul quale è montata la Pancam ruota di 360°, consente di acquisire immagini di tutto l'ambiente circostante e permette un movimento verticale di 180° per poter osservare anche in alto e in basso. Gli occhi elettronici sono grandi quanto il palmo di una mano e permettono di cogliere con grande dettaglio la superficie e il paesaggio marziano (coprono un campo di 16° x 16°) per creare mappe della zona e per individuare le rocce e i terreni da studiare.

Ogni CCD è dotato di una ruota porta-filtri che, opportunamente attivata, permette di osservare con diverse lunghezza d'onda per meglio individuare alcuni tipi di roccia. Inoltre tale sistema di filtri viene anche utilizzato per una importante funzione di orientamento visuale tramite la ricerca di punti di riferimento celesti come il Sole e le stelle. Non dimentichiamo poi che per le trasmissioni con la Terra, l'antenna del rover deve essere sempre puntata nella giusta direzione considerando anche il moto dei due pianeti; e la visione del cielo, così come avveniva per gli antichi navigatori, è un fedele e preciso punto di riferimento naturale ancora oggi utilizzato.

Oltre a sostenere il sistema Pancam, il collo del rover supporta anche un altro importante sensore chiamato mini-TES (Miniature Thermal Emission Spectrometer), uno spettrometro infrarosso che può individuare l'origine mineraria delle rocce attraverso il rilevamento della loro variazione termica. In modo particolare, lo studio dello spettro rilevato permette di conoscere quali materiali rocciosi si sono formati a seguito dell'azione dell'acqua, informazione indispensabile per dare inizio alla ricerca di vita. Il mini-TES è in realtà posizionato all'interno del rover: la testa funziona infatti come un periscopio che tramite un sistema di specchi riflette l'immagine all'interno del collo, alla cui base è posizionato i sensore del mini-Tes.
Con la semplice rotazione della testa e con il movimento di specchi (per guardare in alto e in basso) si evita di muovere un apparecchio molto delicato sia per le proprie caratteristiche costruttive che per l'importante ruolo che riveste nella missione.

La testa contiene inoltre due Navicam, un sistema di navigazione formato da due CCD in bianco e nero con campo di 45° che permettono di rilevare il panorama in 3D e in luce visibile. Questi sensori cooperano con altri 4 CCD detti "Hazcams" (Hazard Avoidance Camera) dotati di un ampio campo di vista (120°) situato verso la parte inferiore del rover per controllare e prevenire rischi durante i movimenti al suolo.

Tra i sistemi di visione del rover ce n'è uno destinato alla visione microscopica. Posizionato sull'estremità del braccio meccanico, il microscopio CCD si può avvicinare ad ogni oggetto da studiare per consentire riprese (altrimenti impossibili) con cui analizzare le proprietà e le caratteristiche delle rocce e del terreno.
Il braccio meccanico è il mezzo con cui il rover manipola e usa i diversi strumenti. Situato nella parte frontale del mezzo, il "robotic arm" può muoversi liberamente grazie ad una spalla, un gomito e un polso sulla cui estremità si trovano 4 strumenti: il microscopio già visto, uno spettrometro per la rilevazione del ferro, un analizzatore a raggi-X per l'analisi della composizione delle rocce e un utensile per l'abrasione con cui viene grattato via il materiale esterno delle rocce per renderne visibile l'interno.

Figura 9: *A sinistra, particolare dell'analizzatore a raggi-X; a destra, uno dei CCD utilizzati sul rover*

Figura 10: *L'utensile abrasivo permette di eliminare gli strati esterni della roccia e consente la successiva analisi degli strati più interni*

Tutte le operazioni del braccio sono supervisionate dalle due Hazcams frontali che permettono di evitare eventuali collisioni del braccio e l'utilizzo degli strumenti a danno dello stesso rover (non vi viene forse in mente la terza legge della robotica di Asimov?)

IL MOVIMENTO

Il rover utilizza 6 ruote da 25 cm, ciascuna delle quali mossa da un proprio motore indipendente; ogni ruota è collegata con un sistema di sospensioni che interagisce con il corpo e che ne garantisce il bilanciamento anche in presenza delle numerose rocce di cui la superficie marziana è tempestata. Con questo sistema il rover potrebbe affrontare abilmente anche pendenze di 45° (in ogni direzione) ma per evitare inutili rischi, il software non autorizza il superamento di 30°.

Sia le ruote anteriori che quelle posteriori girano sul proprio asse verticale per consentire anche la rotazione del mezzo a 360°. Tutti i movimenti sul suolo vengono accuratamente "pensati" e nonostante la velocità di crociera sia davvero bassa (0,18 Km/h = 5 cm al secondo) il solito software previdente "anti rischio" dispone che dopo ogni pochi secondi ci sia uno stop di controllo e di indagine per valutare il terreno e il successivo spostamento. La velocità media scende quindi a 1 cm al secondo, alternando 10 secondi di movimento a 20 di analisi.

L'ENERGIA

L'unica fonte di energia utilizzabile dal rover proviene dal Sole. I computer, i motori, i sensori e tutti gli strumenti di cui è dotato consumano circa 100 W. I suoi pannelli solari forniscono, in piena luce, circa 140 W di potenza e consentono la piena autonomia del robot e la carica di due batterie necessarie per mantenere l'operatività durante le ore di minor luce e durante la notte. Le batterie utilizzate garantiscono la loro piena funzionalità solo per qualche settimana, periodo oltre il quale non si riuscirà più a ricaricarle al 100%. Inoltre, dopo circa 3 mesi, il Sole non sarà più alto nel cielo marziano (si verifica un cambio di stagione anche su Marte) e questo diverso allineamento non permetterà ai pannelli solari di produrre la stessa quantità di energia; anche la polvere della superficie marziana che si depositerà sui pannelli contribuirà al calo di efficienza degli stessi e si riusciranno a garantire soli 50 W...

Il passare dei mesi porterà Marte sempre più lontano dal Sole e l'energia non sarà più sufficiente per mantenere "vivo" il rover...

Figura 11: *Particolare del braccio meccanico e degli strumenti posizionati alla sua estremità*

LE COMUNICAZIONI

Le comunicazioni con la Terra avvengono via radio e sono assicurate da due antenne principali posizionate sul dorso del rover. La prima è una antenna omnidiriezionale low-gain con la quale vengono ricevute e trasmesse informazioni a bassa velocità. La seconda è una antenna direzionale high-gain che trasmette pacchetti di informazioni nella specifica direzione in cui essa è puntata. Questa antenna è orientabile e il computer può muoverla e allinearla con il nostro pianeta senza dover cambiare la posizione del rover. La comunicazione può avviene sia in modo diretto con la Terra che in modo indiretto, tramite i satelliti "2001 Mars Odissey" e "Mars Global Surveyor" che orbitano intorno al pianeta rosso. I satelliti, oltre a compiere le loro missioni specifiche, operano come ponti radio per ritrasmettere i dati ricevuti, una comodità che permette al rover di rimanere sempre in contatto con il centro di controllo anche quando non si trova esposto verso la Terra. Il satellite Mars Global Surveyor, trovandosi nella posizione adatta, ha anche potuto tracciare il

Figura 12: I particolari delle ruote indipendenti del rover in una reale visione artistica e in una foto che ritrae il primo prototipo

processo di atterraggio di Spirit.

Le comunicazioni dirette tra il rover e la Terra (e viceversa) avvengono ad una velocità variabile tra 3500 e 12000 bit al secondo (3,5 K – 12K baud) mentre quelle tra il rover e i satelliti orbitanti, data la relativa vicinanza, operano alla velocità costante di 128 Kbaud. Ogni satellite a causa del proprio movimento in orbita, può essere utilizzato per circa 8 minuti al giorno, periodo nel quale il rover può inviare circa 7,5 MB di dati. Gli stessi dati trasmessi direttamente alla Terra richiederebbero un tempo compreso tra 1,5 e 5 ore.
In ogni caso, Spirit può trasmettere in quest'ultima configurazione per un massimo di 3 ore, tempo fissato da termini di sicurezza per limitare il consumo di energia e per evitare surriscaldamenti eccessivi.

Figura 13: Le due antenne principali del robot: omnidirezionale (a sinistra) e quella direzionale orientabile

IL SOFTWARE

Il rover riconosce le diverse fasi della missione e attiva di conseguenza degli specifici moduli di software. Esistono infatti controlli e priorità diverse in ogni periodo, calcolato dalla partenza della missione all'arrivo sul suolo marziano. Durante le diverse fasi di viaggio le principali funzioni riguardano principalmente il controllo del proprio mantenimento, il check delle condizioni del veicolo e alcune funzioni di comunicazione. I controlli vengono continuamente ripetuti anche durante la fase di ingresso nell'atmosfera marziana, momento in cui gli sbalzi di temperatura devono essere compensati per non danneggiare la tecnologia di bordo (garantita per funzionare da –40 a + 40 °C). Questa parte principale del software può essere considerata la parte di "istinto" che permette di mantenere operativo il rover e che gli consente di sopravvivere grazie ad un feedback che interagisce con le percezioni rilevate dai sensori.
Giunto sul suolo del pianeta, il software provvede a registrare anche tutte le condizioni e le operazioni di base svolte, a gestire la carica delle proprie batterie e alla preparazione di pacchetti di informazioni da inviare durante le fasi di comunicazione con la Terra.

La particolarità del software del rover è quella di poter essere aggiornato anche a missione avviata tramite l'invio di nuovi pacchetti di dati in sostituzione di quelli già installati.
È stata quindi prevista una sezione di base che consente di ricevere ed operare parallelamente al software di gestione vero e proprio. Inoltre, esistono funzioni parallele sempre attive che provvedono a riattivare il sistema principale in caso di eventuale spegnimento dovuto a diversi fattori, come ad esempio i notevoli stress fisici subiti nella fase di ingresso nell'atmosfera e nella "rimbalzante" e imprevedibile fase di atterraggio.

Altre comandi relativi alla missione, alle riprese e all'acquisizione di dati specifici vengono invece impartiti e controllati da Terra direttamente con l'invio dei relativi ordini attraverso le forme di comunicazione già viste.

Figura 14: *Gabbia per 2 colonne*

CONCLUSIONI

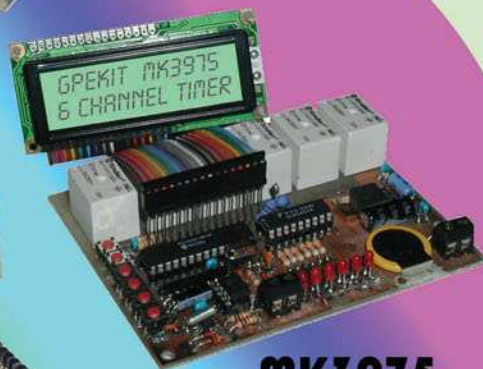
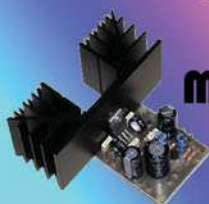
In queste poche righe abbiamo attraversato un mondo nel quale la tecnologia, la scienza, l'esperienza e le capacità umane contano più di ogni altra cosa. Abbiamo immaginato e brevemente rivissuto le insidie di una missione spaziale e analizzato da vicino lo strumento ultimo che concretizza la missione. Abbiamo visto che nulla può essere dato per scontato o lasciato al caso e che spesso le soluzioni più semplici rappresentano la via più sicura per muoversi in un mondo basato sulla precisione millimetrica e sull'importanza di ogni singolo grammo di peso.
Il rover è forse un veicolo robotico più vicino alla nostra realtà quotidiana di quello che potevamo pensare. Dialoga via radio nella banda UHF, trasmette dati come un modem, osserva l'ambiente con dei CCD, utilizza un computer che è già da tempo sulle nostre scrivanie, si muove su ruote e si serve di un semplice braccio meccanico. Tutte tecnologie relativamente note e alla portata degli appassionati più evoluti. Ma il rover è sicuramente diverso e molto più complesso di quello che sembra. In esso sono racchiusi decenni di esperienza, di successi e di errori della storia dell'esplorazione spaziale, decenni di motivazione e di passione da parte di uomini che credono, vivono e muoiono per un solo scopo. Il robot, come dicevamo all'inizio, è solo la punta dell'iceberg così come "$E=mc^2$" è solo una semplice formula. Stranamente, calcoli, studi e analisi talmente complesse da sfuggire alla nostra percezione portano spesso a soluzioni semplici ma geniali.

Nel momento in cui stiamo scrivendo, il rover è già in movimento da qualche giorno sul suolo marziano ed ha inviato numerose immagini e informazioni. Ma, come dicevamo, l'imprevisto è sempre in agguato e dopo uno degli ultimi comandi impartiti via radio dalla Terra, il rover ha confermato la ricezione del messaggio ma non ha poi eseguito l'operazione che gli si richiedeva di svolgere. Per il momento possiamo solo sperare che il nostro lontano e tecnologico esploratore si comporti come altre missioni precedenti, che dopo esitazioni simili, erano tornate a funzionare diventando grandi successi.

Nel frattempo, il Presidente Bush ha annunciato una nuova era spaziale ricca di conquiste, di basi lunari, di sonde automatiche, di uomini su Marte e di consistenti finanziamenti che faranno dimenticare i "giocattoli" fino ad oggi costruiti. Entreremo davvero in quel "futuro" che immaginavamo da piccoli o saranno solo promesse elettorali?
Staremo a vedere.
Per ora, buona fortuna Spirit...

PER SAPERNE DI PIÙ

I siti della Nasa e del JPL (Jet Propulsion Laboratory) sono i punti di riferimento per navigare tra tantissime informazioni di testo, immagini e animazioni. Quelle che seguono sono solo le pagine principali da cui è possibile iniziare l'esplorazione di centinaia di contenuti sempre aggiornati in tempo reale.

- Il punto di partenza con ricche applicazioni interattive on-line per esplorare e saperne di più sulla missione e su Marte: www.nasa.gov
- L'home page del Rover: http://marsrovers.jpl.nasa.gov/home/index.html
- Una raccolta di video e animazioni che descrivono le fasi più importanti della missione e le tecnologie usate: http://marsrovers.jpl.nasa.gov/gallery/video/
- Altre immagini, contenuti multimediali, video e simulazioni: http://athena.cornell.edu

ALIMENTATORE USB
seconda parte

di Maurizio Del corso
m.delcorso@farelettronica.com

Nella prima parte dell'articolo sono stati analizzati tutti gli aspetti teorici in modo da poter comprendere a fondo il funzionamento dell'alimentatore. Ora vedremo la sua realizzazione sopratutto ciò che riguarda il software di gestione per il PC ed il firmware col quale programmare il micro.

IL SOFTWARE PER IL PC

Il software per controllare l'alimentatore tramite il PC è scritto in Visual Basic 6 ed è scaricabile dal sito di FareElettronica sia in forma compilata e pronta per l'installazione, sia in forma sorgente in modo che la possiate modificare a vostro piacimento. La versione corrente è scritta per Win9x e può non funzionare correttamente con Windows2000/NT/XP. All'avvio del programma l'alimentatore dovrà essere connesso, in caso contrario (o in caso di altri eventuali errori) il programma inviterà l'utente a connettere il dispositivo o impostare correttamente la porta seriale. La schermata iniziale è quella di figura1.

Nella sezione "Output" è possibile leggere i valori della corrente e della tensione di uscita aggiornati in tempo reale. La tensione di uscita può essere impostata con passi di circa 1Volt o con passi di circa 0,1Volt se si utilizza la "Regolazione fine". Se invece si intende utilizzare uno dei valori standard (5V, 6V, 9V o 12V) basta selezionare l'opportuno comando.
Dal menu "File" è possibile impostare la porta seriale opportuna, mentre dal menu "Guida" è possibile richiamare un guida in linea che attualmente non è implementata dato che il tutto è spiegato in questo articolo (a voi il compito di aggiungere la guida in linea!). Il funzionamento è molto semplice: all'arrivo di un byte dalla porta USB (la COM fittizia), il programma riconosce se si tratta di un valore di corrente o di tensione, quindi aggiorna il rispettivo display. Se invece viene premuto uno dei tasti per la regolazione, allora viene inviato sul bus USB il byte corrispondente all'incremento/decremento scelto o al valore assoluto selezionato.

IL FIRMWARE PER IL PIC

Anche il programma per il PIC è scaricabile dal sito di FareElettronica sia in formato sorgente che compilato (quindi pronto per essere scaricato nel PIC). Vediamo più in dettaglio quali sono le operazioni che dovrà eseguire il PIC. Il diagramma di flusso del firmware è quello di figura2.

Il programma è costituito quindi da un ciclo infinito che attende, mediante l'uso di un timer, un intervallo di tempo pari a circa 0,5secondi dopodiché legge alternativamente i valori di tensione e di corrente e li invia sul bus USB.
Ma come si può gestire la richiesta di un determinato valore di tensione da parte dell'utente?
Un modo molto elegante per farlo è quello di usare il meccanismo delle interruzioni. Supponete di essere nel vostro ufficio a svolgere il vostro lavoro quando, ad un tratto, vi telefona un vostro collega chiedendovi una informazione. Voi interrompete il vostro lavoro per rispondere al telefono e, una volta soddisfatta la richiesta inoltratavi, mettete giù il telefono e tornate al vostro lavoro.
Questo, a grandi linee, è il funzionamento delle interruzioni: in corrispondenza di un certo evento (la pressione di un tasto di regolazione della tensione sul PC) il programma viene interrotto per eseguire un frammento di codice predefinito, dopodiché l'esecuzione riprende dal punto in cui era stata interrotta.
Per il PIC in questione, il "frammento di codice predefinito" inizia dall'indirizzo 04H (in binario: 00000100), ciò significa che all'arrivo di una interruzione il flusso del programma si inter-

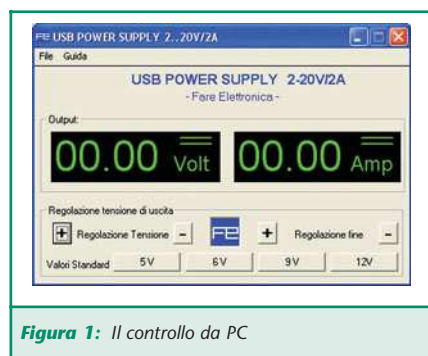
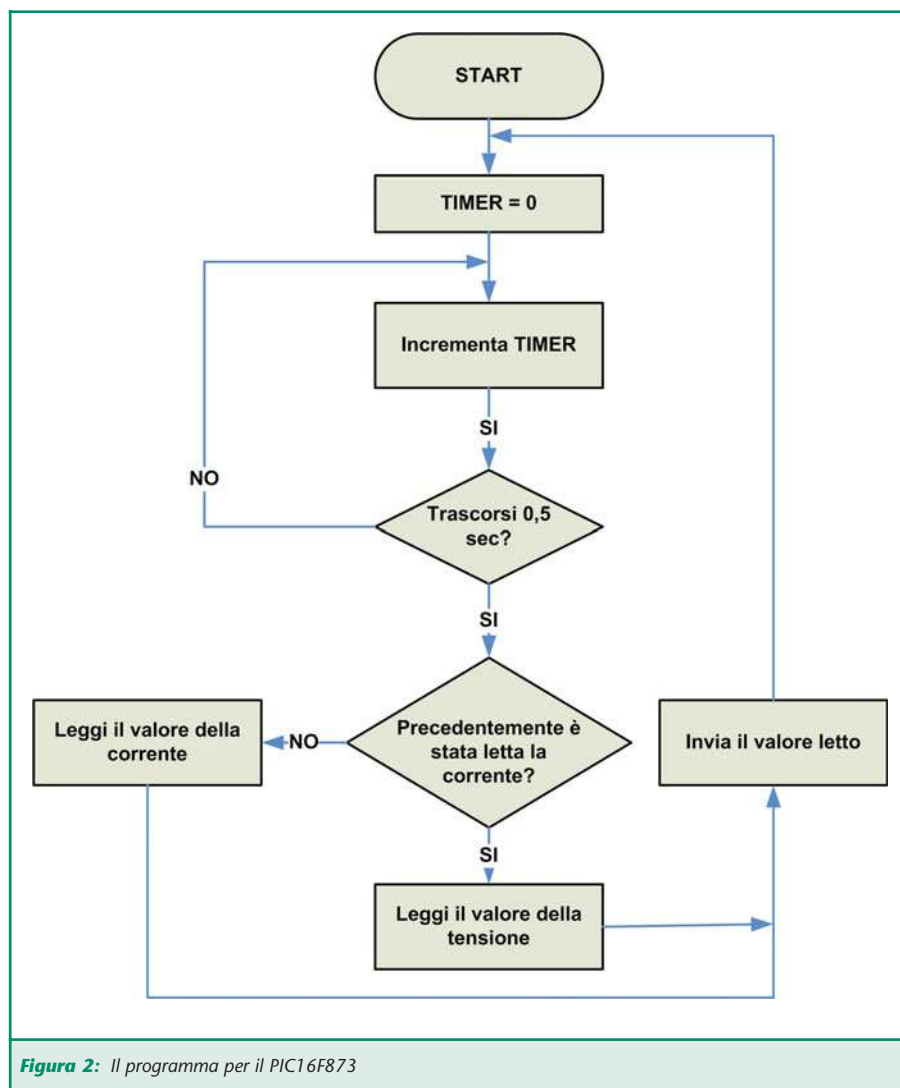



Figura 1: *Il controllo da PC*

Figura 2: Il programma per il PIC16F873

Figura 3: Il circuito integrato LM723

rompe e vengono eseguite le istruzioni a partire dall'indirizzo 04H fino all'istruzione RETFIE che determina la fine della routine di interruzione, quindi la ripresa del programma dal punto in cui era stato interrotto.
Quando un dato arriva dal bus USB, il controllore USB245M manda a livello basso il pin RXF# quindi il piedino RB0/INT del PIC (si veda lo schema di figura4) e si innesca l'interruzione. Il diagramma di flusso della routine di interruzione è quello di figura3.

La prima cosa che viene fatta nella routine di interruzione è bloccare ulteriori interruzioni e salvare tutte le variabili attualmente in uso dal programma (per poi ripristinarle in seguito).
Dunque si va ad eseguire un'operazione di lettura sul USB245M (ponendo RD=0) e si abilita il latch U4 in modo da trasferire il dato al convertitore D/A. Tali operazioni devono essere ripetute finché RXF# non torna al valore alto segnalandoci che non ci sono ulteriori dati da leggere. A questo punto si disabilita il latch, si ripristinano le variabili salvate e si riabilitano le interruzioni.
Nello schema di figura4 si noti la presenza del transistore Q1 e del relè LS1. La loro presenza consente di salvaguardare il transistore di potenza Q3 il quale è sottoposto ad una tensione tra collettore ed emettitore data dalla differenza tra la tensione non regolata (presente al punto A) e la tensione di uscita.
Nel caso si richiedano in uscita tensioni troppo basse e correnti consistenti, la tensione Vce di Q3 viene ad assumere valori elevati e la potenza dissipata potrebbe superare quella massima sopportabile.
Quando si richiedono basse tensioni in uscita (inferiori a 9,5V) il PIC pone a livello basso l'uscita RB5 quindi il relè si trova nello stato di riposo e nel punto A vi è una tensione pari a circa 20V (viene cioè utilizzata solo una parte della trasformatore).
Quando la tensione di uscita supera i 10,5V allora il PIC pone RB5 a livello

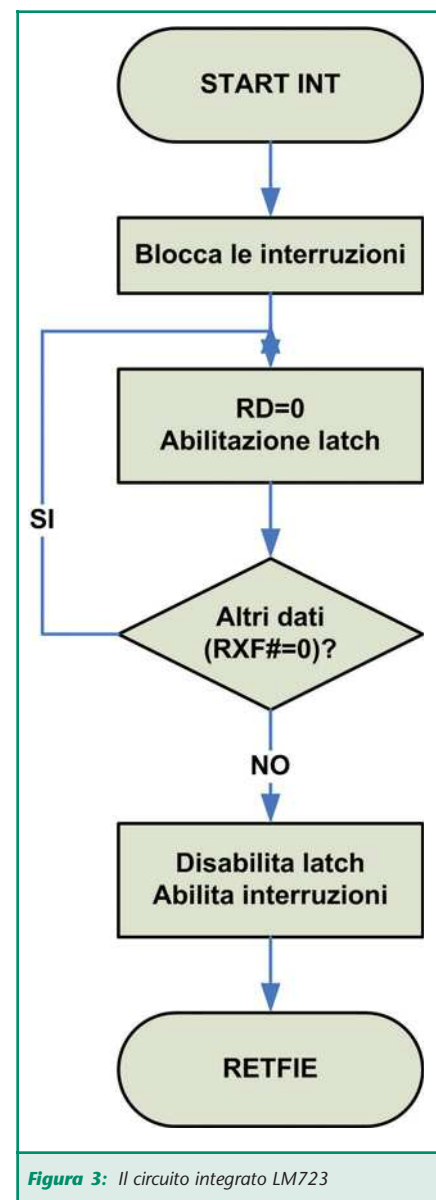
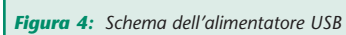



Figura 4: *Schema dell'alimentatore USB*

Elenco componenti	
Sigla	**Valore**
C1	10 nF ceramico
C2	100 nF ceramico
C3,C4	22 pF ceramico
C5,C6	100 pF ceramico
C7,C10	0.1µF 63 V ceramico
C9	47 µF 63 V elettrolitico
C11	4700 µF 63 V elettrolitico
D1	1N4002
D2	Ponte raddrizzatore 400 V 5 A
LS1	Relè 12 V
Q1	BC547
Q2	2N1711
Q3	2N3055
R1, R2, R3	2.2 KΩ 1/4 W
R4, R5, R7, R10÷R13	10 KΩ 1/4 W
R6	0.27 Ω 5 W
R8, R19	1 KΩ 1/4 W
R9	500 Ω 1/4 W
R14	22 KΩ 1/4 W
R15, R17	47 KΩ 1/4 W
R16, R18	470 KΩ 1/4 W
T1	Trasformatore 220/15+15Vac 100VA
U2	LM358
U4	74LS374
U5	PIC16F873
U6	USB245M
U7	LM723
Y1	Quarzo 4 MHz

logico 1 eccitando il relè ed in questo modo nel punto A avremo una tensione pari a circa 42 V. In definitiva Q1 consente di avere bassa tensione di collettore nel caso di basse tensioni di uscita limitando la potenza dissipata da Q3.

TARATURA DEL CIRCUITO

Prima di utilizzare l'alimentatore si deve procedere con l'operazione di taratura per essere sicuri che la tensione di uscita sia effettivamente quella voluta e che il valore visualizzato sia corretto.
Con riferimento allo schema di figura4, mediante i trimmer R19 ed R14 si regolano rispettivamente il valore di corrente e di tensione visualizzato sul PC mentre tramite R9 si regola il valore di tensione realmente generato. Per effettuare la taratura occorre collegare il circuito al PC, avviare il programma e selezionare un valore di tensione (ad esempio 5 V).
Collegando un multimetro all'uscita dell'alimentatore, si dovrà agire su R9 finché il valore letto sul multimetro non coincide con quello scelto dal PC.
Ora si deve agire su R14 per fare in modo che sul PC venga visualizzato lo stesso valore indicato dal multimetro. Come già ampiamente spiegato nella prima parte dell'articolo, il valore della tensione di uscita viene letto dal PIC e convertito in digitale su 7 bit. Ciò significa che se l'uscita può variare da 0 a 20 V, la minima risoluzione visualizzabile è 0,15 V (ovvero 20 V/2^7) per cui può accadere che il valore visualizzato sul PC si discosti leggermente da quello reale.
Per la taratura della corrente si deve impostare dal PC una tensione di 10 V e collegare all'uscita dell'alimentatore una resistenza da 1KΩ (1/2 W) come mostrato in figura 5.

Figura 5: *Schema per la taratura della corrente*

Si noti che il multimetro deve essere usato come amperometro. Sul multimetro si leggerà una corrente di circa 10mA (0,01 A) e si dovrà agire su R19 affinché il valore letto sul PC risulti pressoché uguale a quello indicato sul multimetro. In ogni caso ricordate che i valori di tensione e corrente visualizzati vengono aggiornati alternativamente ogni secondo.

CONCLUDENDO...

Riconosco che la realizzazione di questo progetto non è affatto semplice e, comunque, vi sono ampi margini di miglioramento.
Ritengo però che questo costituisca un forte stimolo alla vostra curiosità e alla vostra voglia di progettare, così potrete migliorare il progetto aumentando la risoluzione del convertitore A/D (magari impiegando un altro PIC o un convertitore A/D del tipo "stand alone"), riscrivere il software cambiandone le caratteristiche, riscrivere il firmware impiegando altre tecniche di programmazione ed aggiungere nuove funzionalità. Potrete anche impiegare frammenti di questo articolo per costruire un acquisitore dati, un voltmetro, un termometro e tutto ciò che suggerisce la vostra fantasia.

Ormai l'USB è alla vostra portata!

MANUALE SEMISERIO DI ROBOTICA

MUOVERE UN BRACCIO ROBOTICO

quinta parte

di Marco Fabbri
marnic@roboitalia.com

Ci siamo impegnati nelle precedenti puntate nel dimensionamento e nelle tecniche per determinare il posizionamento, adesso proviamo a risolvere il problema pratico.

PRENDERE UN OGGETTO IN UN DATO PUNTO

Obiettivo: dato un oggetto in una data posizione dello spazio, calcolare il posizionamento del braccio e delle sue parti per afferrare l'oggetto.
Sembra facile...

Nella figura 1 vediamo il nostro sistema 3D (con origine in O) e un oggetto in rosso che è il nostro target.
Noi abbiamo già il nostro braccio montato su una base semovente (figura 2), la parte nera è la pinza.

Ora noi possiamo muovere la base, ovviamente solo sul piano X-Y, in questo modo porteremo il punto Q in un punto di coordinate X1,Y1,Z1; questa posizione dovrà ovviamente essere tale da avere Xt, Yt, Zt entro il raggio d'azione del braccio (vedremo di approfondire come si calcola il raggio d'azione). Fatto questo è facile dire che se facciamo la differenza tra le coordinate con indice t e quelle con indice 1 otteniamo

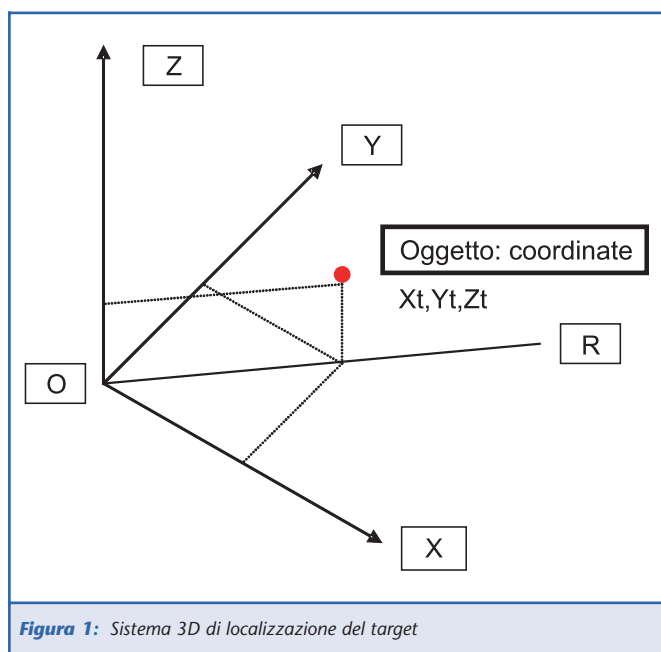



Figura 1: *Sistema 3D di localizzazione del target*

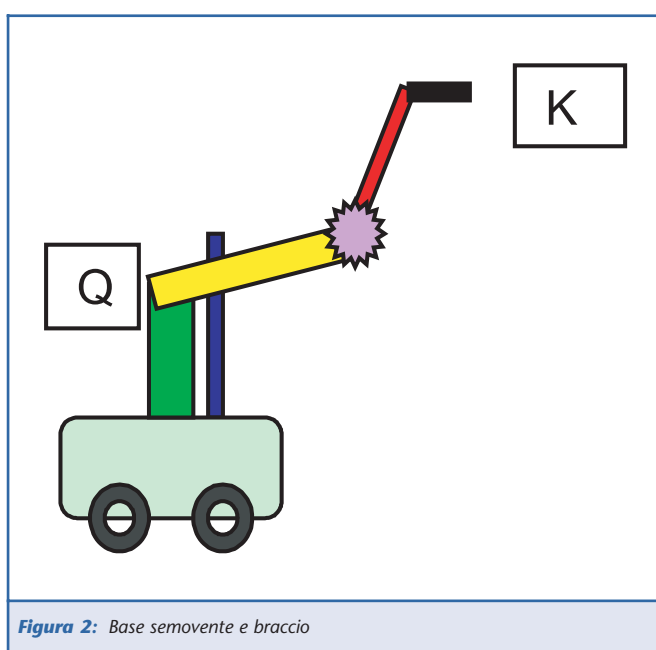



Figura 2: *Base semovente e braccio*

la distanza tra il punto Q di figura 2 e il punto rosso della figura 1, in pratica abbiamo riportato il sistema in modo che Q (figura 2) sia coincidente con O (origine degli assi di figura1).

Comincia la storia degli angoli: la nostra base arriverà nella posizione X1, Y1 con un certo angolo rispetto all'asse X, rappresentiamolo visto dall'alto, figura 3.

Se il nostro punto rosso che è ancora il nostro target ha un angolo ϕ rispetto l'asse X, è facile capire che dovremo ruotare il braccio della differenza dei due angoli, ma quanto vale ϕ?

Intanto avremo:

- Xr=Xt-X1
- Yr=Yt-Y1

Calcoliamo anche la distanza tra il punto rosso (Dpr) e l'origine degli assi in quanto ci servirà:

Dpr=Sqr(Xr^2+Yr^2) (teorema di Pitagora) (Sqr= radice quadrata)

Ora possiamo dire:

ϕ= arccos(Xr/Dpr)

Facendo ora la differenza degli angoli (Fi-Alfa) avremo di quanto ruotare il braccio per portarci nella giusta direzione e abbiamo sistemato il primo dei 4 motori.
Bene, andiamo oltre.
Se tornate alla figura 1, noterete che una volta che ho il braccio nella direzione del mio target (punto rosso) quello che mi rimane da fare è piegare i vari componenti del braccio per raggiungere il punto rosso, in pratica devo lavorare su un semplice piano che è quello determinato dagli assi Z e R. Guardiamo la figura 4.

Le sezioni del braccio sono degli stessi colori del modellino di fig.2, se è chiaro che lavoro solo sul piano Z-R posso quantomeno semplificare il disegno e riportare tutti gli elementi necessari ai fini del calcolo, inoltre per semplificare la cosa ma anche per effettive questioni pratiche posso accettare che la pinza assuma nello spazio solo tre posizioni rispetto l'asse R e cioè quelle visibili in figura 5.

In ordine da sinistra a destra sono rappresentate: presa orizzontale (H), presa verticale dall'alto (Va) e verticale dal basso (Vb), questo non sacrifica le prestazioni in modo eccessivo ma semplifica di molto i calcoli (almeno per me).
Facciamo ora un bel disegno da tenere sottomano (figura 6).

Nello schema rappresentato possiamo fare alcune considerazioni:

- Prima abbiamo calcolato Dpr come proiezione sul piano XY che adesso si chiama Rr.
- Zr sarà semplicemente Zt-Z1.
- a, b, c sono i segmenti del nostro braccio e quindi dati noti.
- Nello schema rappresentiamo la presa H e quindi il punto L avrà coordinate: Zl=Zr e Rl=Rr-c.

Con riferimento alle puntate precedenti di questo manuale quello che ci serve sapere sono gli angoli alfa beta e gamma (α β γ).

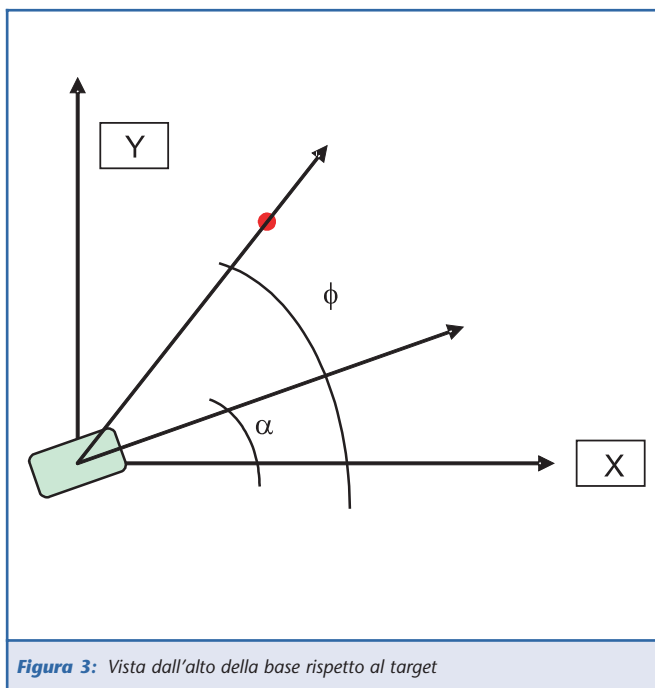



Figura 3: *Vista dall'alto della base rispetto al target*

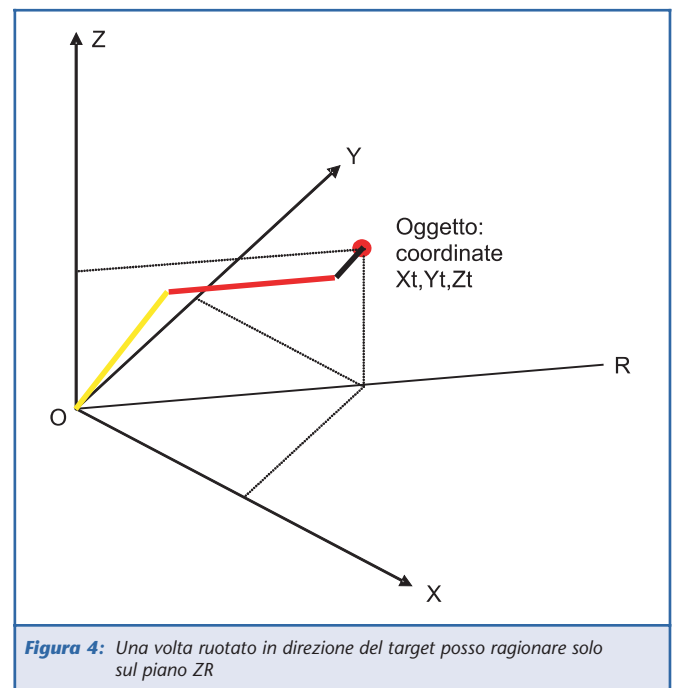



Figura 4: *Una volta ruotato in direzione del target posso ragionare solo sul piano ZR*

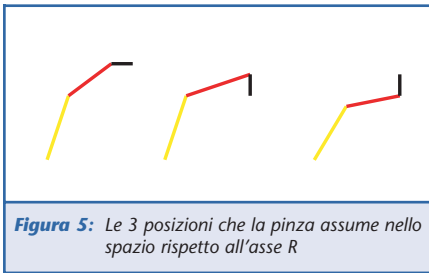



Figura 5: Le 3 posizioni che la pinza assume nello spazio rispetto all'asse R

Cominciamo:

Per prima cosa mi calcolo d, conosco Zl e Rl per cui:

d=SQR(Zl^2+Rl^2)

Ora conosco δ e mi calcolo l'angolo d =arccos(Rl/d).
Quindi abbiamo tutto per calcolarci gli angoli che ci interessano e in particolare avremo che l'angolo QOL (vertice in O e lati OQ e OL) sarà dato da:

QOL=arccos ((a^2+d^2-c^2)/(2*a*d)

Facile ora trovare α dato da:

α = 90-(δ+QOL)

Identica storia per l'angolo OQL:

OQL= arccos((a^2+b^2-d^2)/(2*a*b)

Beta sarà:

β =180- OQL

Ancora più facile per l'angolo gamma, abbiamo infatti che:

α+β+γ=90°

Da cui:

γ = 90-(β+α)

C'è da fare un ragionamento su come si raggiunge la posizione, pensate ad un bicchiere sopra al tavolo, non potete certo arrivarci da sotto, inoltre volendo imitare il movimento umano, l'idea è quella di arrivarci "allungando il braccio" provo con un disegno, figura 7.

Per ottenere questo è sufficiente impostare un iniziale Rr più piccolo di quello calcolato, posizionare il braccio, ripetere la cosa incrementando Rr per passi successivi fino ad arrivare al valore vero.

Figura 6: Grafico degli angoli sul piano ZR

Prese Verticali Va e Vb
Il concetto e la matematica non cambia se non per:

- **Va**
  α+β+γ=180°
  Zl=Zr+c e Rl=Rr
- **Vb**
  α+β+γ=0°
  Zl=Zr-c e Rl=Rr

Gli avvicinamenti saranno fatti seguendo il concetto di prima impostando per Va un valore iniziale maggiore di Zl e decrementando...

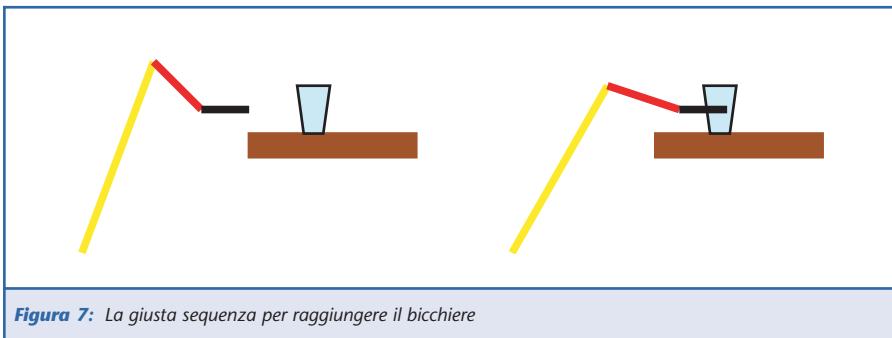



Figura 7: La giusta sequenza per raggiungere il bicchiere

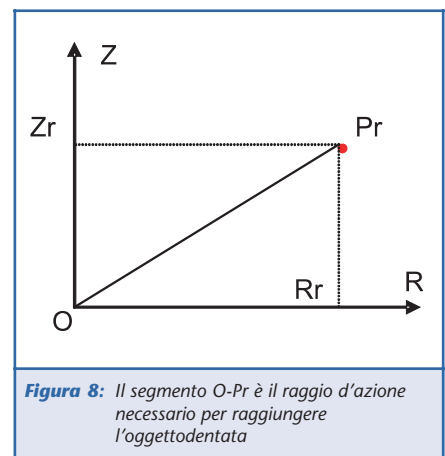



Figura 8: Il segmento O-Pr è il raggio d'azione necessario per raggiungere l'oggettodentata

tandolo per passi successivi, il contrario per Vb.

Il raggio di azione

Inizialmente ho affermato che la base si deve avvicinare al nostro Punto rosso in modo che il braccio riesca a raggiungerlo, aggiungerei in questa fase, raggiungerlo con un buon margine di manovra.

Per fare questo servono ancora due calcoli.

Considerando che il nostro punto rosso è ad una certa quota Z, la distanza maggiore che possiamo avere è quella data dal braccio tutto steso ed inclinato dell'angolo giusto per alzarsi (o abbassarsi) fino alla quota voluta.

Per tenerci un minimo margine non consideriamo la pinza (segmento c).

Ora abbiamo Z1 che è una costante, Zt conosciuto, avremo:

Zr=Zt-Z1
Opr=a+b
Rr=SQR(Opr^2-Zr^2)

Rr era anche prima la distanza della proiezione del Pr sul piano XY, ora questo numero è il raggio della circonferenza che delimita la distanza massima alla quale mi devo trovare, empiricamente potrò stabilire quanto dentro la circonferenza è meglio essere per avere un buon margine di manovra, direi di ridurre Rr di un 30% e procedere per calcolare se sono o no all'interno della circonferenza o raggio d'azione.

Come si fa?

Data una circonferenza di raggio r con centro alle coordinate x y, stabilire se un punto qualsiasi del piano Xa,Ya, è interno alla circonferenza. Questo fatelo voi!

Allego con questa dispensa un foglio Excel che, risolvendo i calcoli, vi mostra il posizionamento dei vari segmenti del braccio. Le celle non sono bloccate o protette perciò siete liberi di modificarlo, i dati del grafico sono sotto al grafico stesso e c'è un foglio con un minimo di indicazioni per la modifica.

È più facile del previsto non essere entro il raggio d'azione del braccio e in questo caso le formule vanno in errore, se poi volete migliorare il foglio, non dimenticate di mandarmene una copia.

BIOQUADRAPOD UN ROBOT A QUATTRO GAMBE

REALIZZAZIONE PRATICA

seconda parte

di Luigi Carnevale
luigi@robot-italy.com

Nella prima parte di questo articolo ci siamo cimentati nella costruzione meccanica del nostro Robot quadrupede. Ora vediamo come creare un semplice programma per farlo camminare

Prima di iniziare desidero riepilogare la situazione: per la logica di comando abbiamo un microcontrollore della famiglia BasicX, precisamente un BX24, montato sulla schedina SimpleBoard; per quanto riguarda la gestione dei servocomandi abbiamo utilizzato lo 8 serial servo controller, della Pololu.
Le due schede comunicano, come avrete intuito, tramite una porta seriale ovvero 2 fili (dati + massa).
La SimpleBoard provvede all'alimentazione della logica delle due schede ed, un apposito circuito di regolazione che utilizza un 7806, fornisce l'alimentazione agli 8 servocomandi.

PERCHÈ IL PROTOCOLLO SERIALE?

Semplice, perchè con soli 2 fili (dati + massa) ci permette di controllare molte periferiche con comandi che possono essere anche complessi. Ci consente, quindi, di creare una "rete" di periferiche che può essere espansa o ridotta senza problemi e senza bisogno di prevedere, a priori, il numero di dispositivi da connettere.
Un altro lato positivo di questo protocollo è che si tratta di uno standard, questo significa che una periferica seriale può essere pilotata, indifferentemente, da un qualsiasi

microcontrollore che rispetti questo standard, ovvero, quasi tutti! Ovviamente anche un personal computer o un palmare possono gestire periferiche seriali.
Utilizzando un cavo in più, quindi due + massa, otterremmo una comunicazione bidirezionale, ovvero la periferica sarebbe in grado di inviarci delle informazioni che potrebbero essere dati oppure dei semplici segnali di conferma o di buon funzionamento del dispositivo.
Nel nostro caso, il secondo cavo non serve poiché questa periferica non prevede l'invio di dati.

IL BX24 E I COMANDI SERIALI

Quanto è complicato gestire dei comandi seriali con un BX24? Questo starà a voi giudicarlo, a mio avviso la flessibilità e la semplicità del BX24 sono eccezionali e difficili da trovare in altri microcontrollori.
Ecco un esempio pratico di come si inizializza la porta seriale COM3:

```
Dim Com3In(1 to 12) As Byte
Dim Com3Out(1 to 12) As Byte
Call DefineCom3(0,5,
  bx0000_1000)
```

Le prime due righe di codice, definiscono due array da 12 bytes cadauno, questi sono il buffer di input ed output. Infatti il BasicX bufferizza i comandi seriali in ingresso e in uscita, questo si traduce in una migliore performance generale delle porte seriali. I buffers devono essere di 9

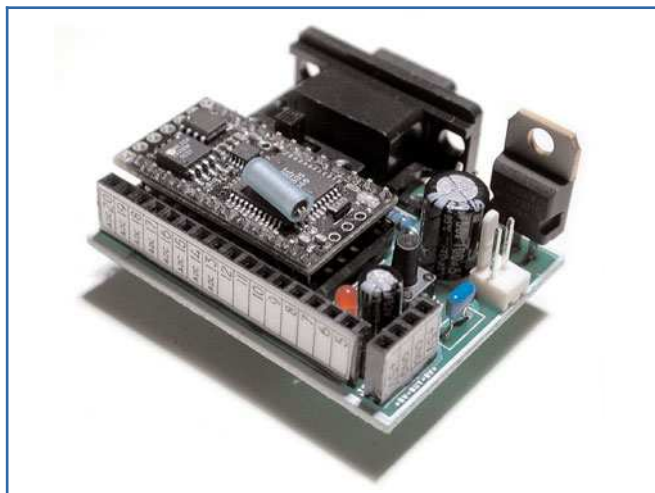



Figura 1: *La SimpleBoard con il BX24 installato*

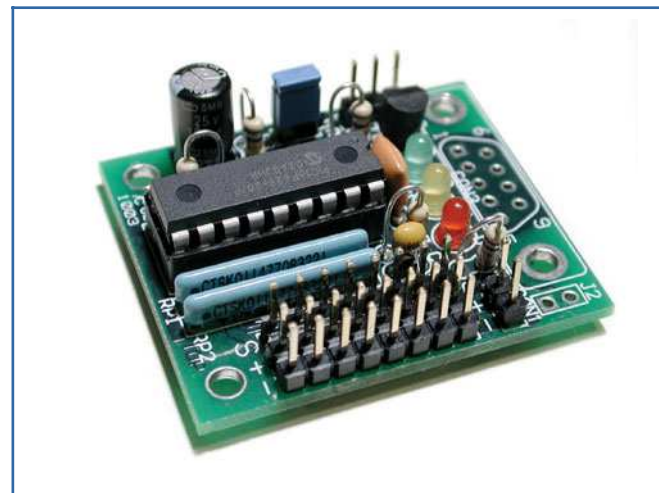



Figura 2: *Il Pololu 8 Serial Servo Controller*

byte + il numero massimo di byte che si invierà in un unico comando seriale. Dal momento che noi invieremo 3 bytes al Servo Controller, il buffer sarà di 12 (nulla ci vieta di farlo più grande).

La terza riga, definisce la porta COM3, questa può essere impostata su due qualsiasi dei pin disponibili sul BX24 dal 5 al 20. La sintassi del comando richiede: pin d'ingresso, pin di uscita, parametri della porta. Se non serve di inviare o di ricevere comandi seriali il rispettivo pin si può impostare a zero risparmiando una porta I/O del microcontrollore. Nel nostro caso ho impostato il pin d'ingresso a 0, in quanto non ci serve ricevere dati, il pin di uscita è il 5. I parametri richiesti dal controller Pololu sono 8N1, per impostare tali parametri si deve inserire il codice bx0000_1000 (come da tabella fornita nella documentazione del BasicX).

Figura 3: *Lo schema elettrico del circuito per la determinazione della posizione degli 8 servocomandi*

Questo va fatto solo ad inizio programma, oppure ogni volta che si intendono cambiare i pin della COM3 o la dimensione dei buffers. Ora non ci resta che aprire la coda (il buffer) con il comando OpenQueue:

```
Call OpenQueue(Com3In,12)
Call OpenQueue(Com3Out,12)
```

Ed infine, aprire la porta, impostando la velocità a 9600 baud, come richiesto dal servo controller:

```
Call OpenCom(3,9600,Com3In,
  Com3Out)
```

Da adesso in poi, per inviare un qualsiasi comando seriale, dalla porta COM3, è sufficiente utilizzare il comando PutQueue oppure PutQueueStr se intendiamo inviare delle stringhe di testo.
Un comando tipico che possiamo inviare al controller Pololu, potrebbe essere:

```
Call PutQueue(Com3Out, 255,
  2, 120)
```

Questo semplice comando, ordinerà

al Serial Servo Controller, identificato con il numero 255, di muovere il servo numero 2 nella posizione 120 che rappresenta, più o meno, la posizione centrale del servo. Il controller continuerà a generare l'impulso PWM e il servo manterrà questa posizione fino a nuovo comando.

Determinare la posizione dei servi

Prima di passare allo studio del funzionamento della camminata del quadrupede, dobbiamo trovare un sistema per determinare, in termini numerici (da 0 a 255), la posizione di ognuno degli 8 servocomandi.

Dal momento che non tutti i servi sono identici tra loro, sicuramente i valori da inviare saranno diversi per ogni servo, un servo sarà centrato con un valore di 120, un altro servo invece con 105; lo stesso discorso vale per tutte le varie angolazioni che dovremo impostare per far camminare il Robot.

Per raggiungere questo scopo, potremmo collegare il Pololu SSC direttamente al PC e con un programma di comunicazione seriale, tipo Hyperterminal, inviare i singoli comandi seriali per impostare i vari servi.

Su Internet ci sono diversi programmi di comunicazione seriale, anche specifici per il Mini SSC che è perfettamente compatibile con il Pololu.

Gli utenti più esperti in programmazione su PC potrebbero creare delle interfacce personalizzate con manopole e cursori per muovere i servocomandi; poi se li voleste pubblicare su www.robot-italy.net farete la felicità di molti utenti di questa community.

Un'alternativa, potrebbe essere quella di utilizzare un commutatore rotativo a 8 posizioni, 2 pulsanti e 10 resistenze da 1 KΩ, per costruire un piccolo dispositivo che ci permetta di incrementare e decrementare la posizione dei servi con i pulsanti e, tramite il commutatore, selezionare il servo da impostare. Il commutatore può anche essere sostituito con un dip-switch a 8 bit.

Tramite la COM1 del BX24, collegata al PC, potremo sapere in qualsiasi momento la posizione esatta di ogni servo e quindi impostarla durante la programmazione. In Figura 3 trovate uno schema che rappresenta il circuito, lo potete realizzare con una schedina millefiori da piazzare momentaneamente sul Robot.

In pratica, con il commutatore si seleziona il servo, con i pulsanti si incrementa o decrementa il valore, provocandone la rotazione in senso orario o antiorario, quando abbiamo raggiunto la posizione desiderata, controlliamo sullo schermo del PC il valore e prendiamo nota.

All'inizio è un lavoro abbastanza noioso ma dopo un po' ci si prende dimestichezza e si impara a stimare le posizioni e quindi è molto più rapido e facile.

Naturalmente per poter funzionare il dispositivo richiede una programmazione. Il programma necessario, in realtà, è lo stesso che occorrerà per far camminare il Robot, con alcune lievi modifiche. D'altronde il bello di lavorare con i BasicX è proprio questo: dall'idea alla realizzazione ci vogliono pochi minuti. Ovviamente il BasicX può implementare anche compiti molto complessi ed in questo caso, probabilmente, ci vuole qualche ora, ma è niente se paragonato alle giornate e soprattutto nottate, che occorrono con linguaggi complessi come l'assembler e spesso anche il C.

Di seguito illustro soltanto le modifiche da apportare al programma principale di cui parleremo in seguito.

Innanzitutto ci creiamo una costante per ogni servo a cui daremo il valore del pin al quale è collegato il commutatore.

```
Const Sv00 As Byte = 20
Const Sv01 As Byte = 19
Const Sv02 As Byte = 18
Const Sv03 As Byte = 17
Const Sv04 As Byte = 16
Const Sv05 As Byte = 15
Const Sv06 As Byte = 14
Const Sv07 As Byte = 13
```

In seguito andiamo a creare altre due costanti per i pulsanti di incremento (UpBtn) e decremento (DwBtn).

Quando viene premuto un pulsante, il relativo pin del BX24 diventa alto quindi vale 1, viceversa, in posizione di riposo varrà 0.

```
Const UpBtn As Byte = 12
Const DwBtn As Byte =11
```

Ora ci dobbiamo creare una routine che vada a controllare lo stato di questi pulsanti e, nel caso vengano premuti, impostare un nuovo valore di posizione del relativo servo; presto fatto:

```
Do
  if GetPin(UpBtn) = 1 then
    Call IncrSub()
    Call Delay(0.05)
  End if
  If GetPin(DwBtn) = 1 then
    Call DecrSub()
    Call Delay(0.05)
  End if
Loop
```

In questo codice troviamo un ciclo Do..Loop, in pratica viene eseguito il codice che segue l'istruzione Do fino ad arrivare a Loop che rimanda all'istruzione Do, ripetendo il ciclo all'infinito.

La seconda istruzione è un classica condizione If..Then..End If, le istruzioni che seguono il Then saranno eseguite se si verifica la condizione If.

Nel nostro caso se il pin UpBtn (pin 12 del BX24) è a livello alto (il pulsante è premuto) allora esegue le istruzioni successive, ovvero: Call IncrSub() che è la routine che incrementa il valore (la posizione) del servocomando, come vedremo in seguito, e Call Delay(0.05) che è un ritardo di 0.05 secondi. Il ritardo viene applicato per evitare che la pressione del pulsante sia interpretata come una "serie" di pressioni, provocando un risultato falsato che nel gergo tecnico viene chiamato "bouncing", il ritardo rappresenta, appunto, il "debounce".
Se, invece, il pin UpBtn è a livello basso (il pulsante non è premuto) l'istruzione passa direttamente alla fine, End If, e quindi alla successiva istruzione.
La condizione seguente fa la stessa cosa solo che controlla il pulsante DwBtn e lancia la routine di decremento del valore.
Ora non resta che scrivere le routine di incremento e decremento:

```
Sub IncrSub()
```

Se ci troviamo all'interno di questa routine è perchè dobbiamo incrementare il valore di uno o più servocomandi (nel caso del dip-switch), ma quali sono i valori da incrementare?
Dobbiamo verificare quali sono i pin a livello "uno". Lo stato dei pin viene messo a livello alto dal commutatore rotativo o dai DIP Switch.
Iniziamo con un ciclo condizionale If..Then..End If, partendo dal Servo 0:

```
If GetPin(Sv00) = 1 then
```

Se il pin Sv00 (ingresso 20 del BX24) è a livello alto:

```
Sv00Val = Sv00Val + 1
```

Allora il valore della variabile (di tipo Byte) Sv00Val è incrementata di 1 (nel programma vero e proprio la variabile è stata inizializzata con l'istruzione: Dim Sv00Val As Byte).

```
Debug.print "Pos servo 0 = ";
  Cstr(Sv00Val)
```

Ora scriviamo sullo schermo il nuovo valore di posizione del servo 0: il comando Debug.Print invia una stringa direttamente alla porta seriale COM1, la quale non ha bisogno di essere inizializzata, in quanto è una porta fisica.
La COM1, nel nostro caso, è collegata al PC che sta eseguendo un programma di comunicazione seriale tipo Hyperterminal oppure il programma proprietario dei BasicX che, a sua volta, contiene una finestra terminale.
Quindi questo comando invierà il testo contenuto all'interno delle virgolette, poi il valore della variabile Sv00Val opportunamente convertito in stringa (era Byte) dal comando Cstr(). Il punto e virgola serve a separare le stringhe da inviare alla seriale.

```
Call SMove(8, Sv00Val)
```

Non ci resta che muovere il servo, mediante la routine SMove, per visualizzare nel mondo reale la posizione fisica del servocomando. La routine SMove, che vedremo in seguito, richiede il numero del servo (8) e la posizione (Sv00Val). In questo caso il servo 0 viene chiamato 8 per un motivo molto semplice: lo standard del MiniSSC prevede la definizione dei servi da 0 a 7 per un ampiezza di movimento del servo di 90° e da 8 a 15 per un'ampiezza di 180°, a noi servirà la seconda.

```
end if
```

Il primo ciclo condizionale è terminato, ora dobbiamo verificare se ci sono altri servi da incrementare. Naturalmente se la condizione precedente non si fosse verificata, ovvero, se il pin Sv00 fosse stato a livello basso, ci saremmo immediatamente ritrovati a questo punto.

```
If GetPin(Sv01) = 1 then
  Sv01Val = Sv01Val + 1
  Debug.print "Pos servo 1 =
    "; Cstr(Sv01Val)
  Call SMove(9, Sv01Val)
end if
...
```

La cosa si ripete per tutti e 8 i servi.
La routine che segue è quella di decremento, in pratica fa le stesse cose della precedente ma sottrae un valore anziché addizionarlo.

```
Sub DecrSub()
  If GetPin(Sv00) = 1 then
    Sv00Val = Sv00Val - 1
    Debug.print "Pos servo 0 =
      "; Cstr(Sv00Val)
    Call SMove(8, Sv00Val)
  end if
  ...
End Sub
```

Ora vediamo la routine che ci permette di inviare i comandi seriali al controller e quindi di muovere i servi:

```
Sub SMove(_ByVal ServoNum as
 Byte, ByVal ServoVal as Byte)
Dim ServoParam(1 to 3) as Byte
  ServoParam(1) = 255
  ServoParam(2) = ServoNum
  ServoParam(3) = ServoVal
  Call PutQueue(Com3Out,
   ServoParam, 3)
End Sub
```

Iniziamo con la prima riga:

```
Sub SMove(_ByVal ServoNum as
 Byte, ByVal ServoVal as Byte)
```

Per muovere i servi ci occorrono due parametri: numero del servo da muovere e posizione, quindi la rou-

tine si aspetta due variabili di tipo Byte: ServoNum e ServoVal.

```
Dim ServoParam(1 to 3) as
 Byte
```

Dimensioniamo una variabile array di tipo Byte con 3 posizioni.

```
ServoParam(1) = 255
ServoParam(2) = ServoNum
ServoParam(3) = ServoVal
```

Assegniamo ora i valori all'array: il primo ha valore fisso 255 che è il codice di inizializzazione del MiniSSC, il secondo è il numero del servo ed il terzo è il valore del movimento.

```
Call PutQueue(Com3Out,
 ServoParam, 3)
```

Una volta che l'array è definito possiamo inviarlo alla porta seriale COM3 con un semplice comando PutQueue(porta com, variabile array, numero di valori da inviare).

A questo punto abbiamo tutti i dati essenziali per scrivere il programma completo ed iniziare a determinare la posizione dei servi, i limiti, le varie posizioni, eccetera.

Con questo sistema è possibile ed utile determinare i valori di limite di ogni servo. E' infatti sconsigliabile impostare un valore che va oltre al limite fisico del servo, si rischia di danneggiarlo.
Testate quindi tutti i servi partendo da 0 fino a 255 per trovare il limite superiore e inferiore e prendetene nota, quei valori non devono essere mai superati. Per limite si intende l'ultimo valore che fa muovere il servo, per esempio: impostando 15 come valore, il servo va quasi tutto da una parte, decrementando il valore il servo fa un ulteriore scatto verso la stessa parte, quando il servo non si muove più, nonostante sia stato decrementato il valore, avete trovato il suo limite inferiore. Il limite superiore funziona allo stesso modo ma bisogna incrementare il valore.

TEORIA DELLA CAMMINATA

La teoria della camminata dei quadrupedi è molto vasta e anche piuttosto complessa, cerchiamo di riassumerla molto sinteticamente poiché va fuori dallo scopo di questa realizzazione.
Per mantenere una posizione di equilibrio, è necessario che almeno tre gambe tocchino il suolo e il COG (centro di gravità) sia opportunamente distribuito sulle tre gambe. Ogni movimento che andremo a fare deve necessariamente rispettare questa regola, altrimenti il nostro Robot cadrà inesorabilmente.
Consiglio di curare attentamente la distribuzione del COG del Robot, eventualmente regolandolo con l'aggiunta di pesi di bilanciamento, per determinare dove posizionarli è sufficiente fare delle pose statiche con il Robot ed osservarne la stabilità, quindi correggere di conseguenza.
Per muoverci dovremo sollevare una gamba, avendo cura di spostare il COG sulle altre tre, prima di staccare la gamba dal suolo, in questo modo ci garantiremo una situazione di equilibrio.
Un sistema, forse più facile, di far camminare il nostro Robot, è la camminata veloce. Infatti, in questo caso, la distribuzione dei pesi risulta meno critica, quindi è probabilmente la strada migliore per iniziare, facendo attenzione, prima di fermarci, di avere sempre tre gambe ben poggiate al suolo e il COG correttamente distribuito.

IN PRATICA

Ho cercato di schematizzare in figura 4 la camminata che ho messo a punto per il nostro Legged, sicuramente non è perfetta e andrebbe perfezionata, però ci

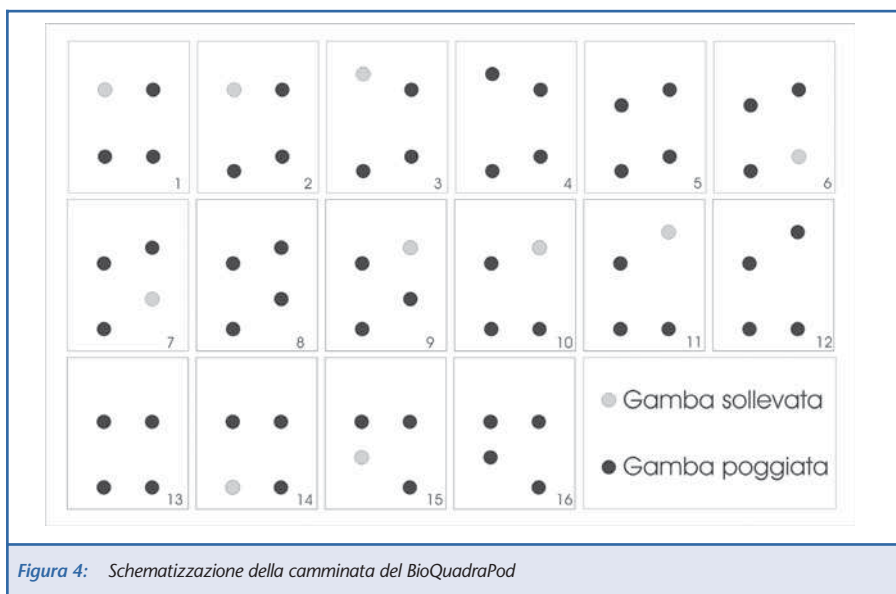



Figura 4: *Schematizzazione della camminata del BioQuadraPod*

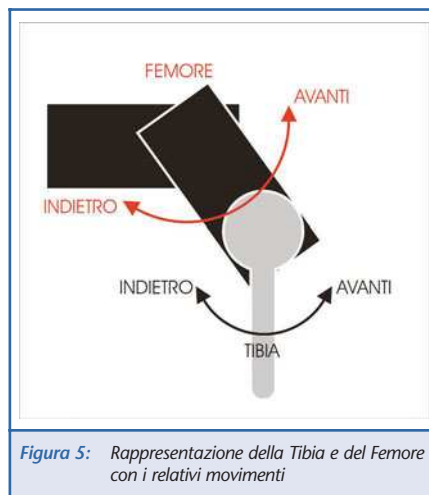



Figura 5: *Rappresentazione della Tibia e del Femore con i relativi movimenti*

permette di camminare e quindi lo scopo è stato raggiunto.
Si sono resi necessari 16 passi di programma per ottenere una camminata omogenea e funzionale, partendo da una posizione centrale dei servi.

Per poter realizzare il programma in modo più semplice ho distinto gli arti in Femore e Tibia e ho assegnato le denominazioni agli 8 servi, come raffigurato nelle figure 5 e 6.

I servi sono stati denominati utilizzando le sigle LF, RF, LR, RR che significano Left Front, Right Front, Left Rear, Right Rear; voi naturalmente potete cambiare liberamente denominazione se vi rimane più semplice.

LA PROGRAMMAZIONE

Vediamo ora un esempio di programmazione, ovviamente per motivi di spazio non troverete tutto il listato, ma potrete scaricarlo dal sito www.robot-italy.net nell'area Downloads.
Grazie al programma precedente, siamo stati in grado di trovare i limiti dei servi e la posizione centrale, in questo modo possiamo facilmente determinare, in modo approssimativo, tutte le posizioni intermedie per poi aggiustarle provando il Robot.
Ci creeremo una routine chiamata Walk che provvederà a settare i servi,

LF RF

LR RR

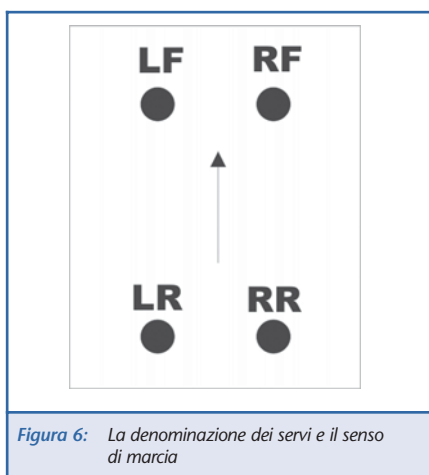



Figura 6: *La denominazione dei servi e il senso di marcia*

uno ad uno, per ottenere la camminata: prima impostiamo il valore della variabile, poi chiamiamo la routine SetSvPos() che andrà ad eseguire il comando e infine introdurremo un piccolo ritardo, per passare poi al passo successivo. Il ritardo andrà regolato per ottimizzare la camminata, a volte può anche essere omesso, dipende da molti fattori.

```
Sub Walk()
   'LF femore avanti
   Sv00Val = 50
   Call SetSvPos()
   Delay(0.1)

   'LF distende tibia
   Sv01Val = 46
   Call SetSvPos()
   Delay(0.1)

   'LF femore indietro
   Sv00Val = 128
   Call SetSvPos()
   Delay(0.1)
   ...
End Sub
```

Ora vediamo la routine SetSvPos():

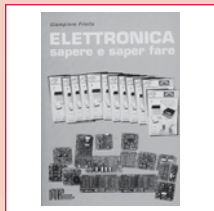
```
Sub SetSvPos()
   'Servo 0
   Call SMove(8, Sv00Val)

   'Servo 1
   Call SMove(9, Sv01Val)
   ...
End Sub
```

Questa routine richiama la routine Smove() che abbiamo analizzato prima, in pratica imposta ciclicamente la posizione di tutti i servi in base ai parametri delle variabili.
Fate bene attenzione che questo programma è stato fatto volutamente in questo modo per facilitare lo studio e l'implementazione della camminata e delle altre posizioni, una volta trovati gli algoritmi giusti, può risultare conveniente semplificarlo ed ottimizzarlo.
È anche possibile eseguire delle posizioni prestabilite, come ad esempio seduto oppure sdraiato, memorizzando la posizione e dandogli un nome, nell'esempio che segue, ho memorizzato la posizione centrale di tutti i servi, quando il Robot si deve fermare è sufficiente chiamare questa funzione per raggiungere una posizione di stabilità:

```
Sub CenterPos()
   Sv00Val = 100
   Sv01Val = 51
   Sv02Val = 93
   Sv03Val = 120
   Sv04Val = 93
   Sv05Val = 88
   Sv06Val = 102
   Sv07Val = 144
   Call SetSvPos()
End Sub
```

Di queste routine è possibile farne quante ne volete e sono molto utili in quanto possono essere determinate con cura e provate a fondo prima di memorizzarle, in seguito si richiamano in un attimo.

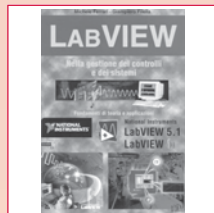
CONCLUSIONI

Far camminare un quadrupede non è molto semplice, ma con la pazienza ci si riesce e la soddisfazione è grande. Questa particolare configurazione delle gambe, oltre a rappresentare una soluzione molto economica, ci permette di fare una gamma di movimenti vastissima, è sicuramente molto interessante sperimentare gli innumerevoli atteggiamenti di questo Robot biomorfico.
Se volete un Robot più stabile potete sempre aggiungere 4 servocomandi, utilizzando un contenitore più lungo e un Serial Servo Controller da 16 servi. Con 6 gambe, l'equilibrio e la stabilità sono garantiti, come pure la capacità di superare pendenze più ripide senza ribaltamenti.

ELETTRONICA
SAPERE E SAPER FARE
G. Filella
Questo libro é dedicato a chi ha un'innata passione per questa scienza. Molti gli argomenti trattati, dalla saldatura, circuiti stampati, filtri oscillatori, amplificatori.
Pagine 228 **cod. 1005 - €12.80**

ELETTRONICA DIGITALE
SAPERE E SAPER FARE
G. Filella
Dopo il volume dedicato all'Elettronica di base è disponibile il nuovo volume dedicato all'Elettronica digitale. Come per il suo gemello è possibile abbinare teoria e pratica con i kit dimostrativi.
Pagine 160 **cod. 2089 - €12,80**

SMD
COMPONENTI E TABELLE DI CONFRONTO
Giancarmelo Moroni
Componenti SMD. Contenitori e utilizzo. Resistenze SMD. Lettura. Codifica. Trimmer SMD. Condensatori SMD. Ceramici. Film. Poliestere. Elettrolitici. Filtri rete. Connettori SMD. Fusibili. Interruttori. Zoccoli. Led. Transistor. Tabella comparativa.
Pagine 111 **cod. 3426 - €9,00**

RIPARARE
MONITOR PER PC ALIMENTATORI A COMMUTAZIONE GRUPPI DI CONTINUITA'
E. Fico - A. Deotto
Guida pratica per capire il funzionamento e imparare i trucchi, le scorciatoie, le tecniche indispensabili per una rapida riparazione.
Pagine 159 **cod. 3162 - €16,60**

CASSE ACUSTICHE
L'Autocostruzione
In un unico CD-ROM la guida al progetto ed autocostruzione dei diffusori acustici. Il suono. La catena audio. L'altoparlante. I fenomeni acustici. La cassa chiusa. Il bass reflex. Sistemi a più altoparlanti. Sistemi caricati a tromba. Filtri cross-over. I componenti. Esempi di calcolo.
Completamente in italiano. Un'opera completa, di facile consultazione, che descrive i diversi aspetti della tecnologia del suono applicati alla realizzazione delle casse acustiche nelle varie tipologie.
CD-ROM **cod. 4283 - Euro 34,49**

LABVIEW
Nella gestione dei controlli e dei sistemi
M. Ferrari - G. Filella
LabVIEW è un rivoluzionario ambiente di programmazione la cui struttura grafica ad oggetti consente di controllare dei dispositivi esterni, siano essi strumenti commerciali o schede interfaccia. Questo volume propone una serie di moduli che consentono di approfondire gli aspetti teorici e applicativi correlati all'utilizzo di LabVIEW. Per questo motivo tutti gli argomenti trattati sono corredati da esempi ed esercizi che hanno anche lo scopo di evidenziare i possibili tranelli legati ad un tipo di programmazione non tradizionale.
Pagine 121 **cod. 3216 - €17,50**

PIC BOOK 2
Con CD allegato
G. Galletti
Naturale proseguimento del primo volume, questo libro completa la panoramica sui PIC. Andremo ad utilizzare sempre il 16F84 ma anche il 16F876, il 12C508 ed il 12CE674 due 4+4 pin dalle caratteristiche a dir poco sorprendenti. La novità principale è che quì parleremo di esperimenti oltre che di progetti. Nel CD accluso il lettore troverà tutto quello che gli serve, compresi i vari programmi nonchè tutti i listati presenti in cartaceo nel volume.
Pagine 170 **cod. 3079/2 - €23,50**

PIC BOOK
Dischetto allegato
G. Galletti
PICBOARD: Un circuito mille fori un po' speciale - TIMEPIC: Timer per conteggi alla rovescia - BLINK: Lampeggiatore per motociclo - SEMAPIC: Piccolo semaforo per modellisti - BELL: Campanello apri porta - ALARMPIC: Piccola barriera per i ladri di appartamento - PICMUSIC: Carillon programmabile - VIDEOPIC: Commutatore video con scanner - GESTIONE LCD: HITACHI 44780 a 4 bit - Macro - Le tempistiche del Micro - Interfacciare il PIC 16x84 - Alcuni siti WEB che trattano i PIC - ARIANNA: Come tessere circuiti stampati - Set istruzioni PIC 16C84.
Pagine 96 **cod. 3079 - €18,00**

LA CORRETTA REALIZZAZIONE DEI SISTEMI FOTOVOLTAICI
G. Rodolfi
Nozioni di elettrotecnica generale. Nozioni di impiantistica applicata. Sistemi fotovoltaici, pannelli fotovoltaici, dimensionamento, installazione, orien-tamento, collegamenti el, note di manutenzione dei pannelli, batterie di accumulatoti, centralina di controllo, macchine statiche di conversione elettrica. Quadro di controllo. Dispositivi utilizzatori, corpi illuminanti, frigoriferi e congelatori. Sollevamento acqua, elettropompe.
Pagine 96 **cod. 3218 - €11,50**

IL PLC
Nella gestione dei controlli e nell'automazione
G. Filella
Al volume è allegato un software, sviluppato in ambiente Windows, che trasforma il vostro PC in un PLC virtuale, per apprendere la logica Ladder attraverso l'animazione di alcuni automatismi. Nel testo vengono messi a confronto i PLC di più recente diffusione prodotti da Siemens, Omron e GE. I principi di funzionamento del PLC. Le memorie. I moduli di ingresso e di uscita. L'elaborazione di un programma all'interno del PLC. Le tecniche di programmazione e la Normativa IEC1131-3. Unità didattiche per la progettazione e schede guida.
Pagine 122 **cod. 3215 - €18,00**

ONDE RADIO E TELEGRAFIA IN MARE
Con Racconti di Marconisti a Bordo
U. Cavina
Wireless e marine da guerra.
Radiogoniometro, uno strano percorso.
Radiotelegrafisti nelle marine da guerra.
Radiotelemetro e radar. Onde radio in immersione. Radiocomunicazioni e sommergibili: una difficile convivenza. Moderni sistemi di radiocomunicazione. Morsemare. La telegrafia aeriforme (aerografia). Origine delle abbreviazioni radiotelegrafiche e del codice "Q". Morse e onde corte. Marconisti a bordo.
Pagine 183 **cod. 4296 - Euro 15,00**

LA RIPARAZIONE DEI TELEVISORI A COLORI
R. Salerno - A. Deotto
Generalità sul TVC. Sezione RF e FI. sintonia e memoria. Il microcomputer. Teletext. Audio FI e Finale BF. Sezione colore, finali colore. Sezione sincronismi e deplessione verticale. Deflessione di riga, generazione AT e EAT. Correzione E-O. Alimentatore. Varie. Appendice. Indice dei guasti proposti e commentati.
Pagine 207 **cod. 3264 - €17,60**

LA RADIO A GALENA: ORIGINI E PROGETTI
Giancarmelo Moroni
Studi sulla galena. Come funziona il detector a cristallo. Sull'isolamento degli aerei. Montaggio di un antenna interna. La costruzione dei trasformatori di alta frequenza. Bobine a solenoide con prese intermedie. I cristalli oscillanti. Apparecchio supercristallo R.T. 19. Tre classici apparecchi a galena. Un amplificatore a microfono magnetico per ricevitore a cristallo. Un nuovo ricevitore a cristallo. Un apparecchio a cristallo molto efficiente. La "Crisolina". Il "Multiplex". Il Galenofono II. Il Bigristallofono. Il Cristallovox. Il "Selettivissimo". Il "Galenofono III". Il "Dui-bigri-galenofono". C.R. 511: ricevitore a cristallo perfezionato per la ricezione a grande distanza. Il "Galenovariometro". Nomogramma per il calcolo dell'induttanza di una spira circolare (E).
Pagine 138 **cod. 3282 - €13,50**

CAVI CONNETTORI E ADATTATORI
La più completa banca dati PC le connessioni PC-audio-video
A. Casappa
Il manuale comprende le connessioni per computer di vari tipi come AMIGA, C64, Spectrum, IBM Compatibili, Audio-Video, IDE, SCSI, ETHERNET, PCMCI, monitor, modem e quant'altro sono riuscito a reperire. Per una migliore consultazione, ho suddiviso il volume in tre parti; • Adattatori • Connettori • Cavi di collegamento
Pagine 80 **cod. 2099 - €10,33**

LA RIPARAZIONE DEI VIDEOREGISTRATORI
R. Salerno - A. Deotto
Generalità sul VCR. Il front-end. La sezione video, percorso dei segnali di luminanza e crominanza in registrazione. La sezione audio. L'alimentatore. La gestione della parte meccanica. Il sistema di controllo. I circuiti servo. La meccanica. Dalla bobina datrice al perno guidanastro di sinistra. Drum testine video e perni guidanastro. Dal perno capstan alla bobina di raccolta. Indice dei guasti proposti e commentati.
Pagine 173 **cod. 3427 - €16,60**

VISITA IL NOSTRO NUOVO SITO INTERNET
www.sanditmarket.it

Oltre 1000 titoli

- ELETTRONICA
- ROBOTICA
- AUTOMAZIONE
- TELEFONIA
- STRUMENTI DI MISURA
- RADIOTECNICA
- TV - RADIORIPARAZIONI
- EQUIVALENZE- DATA BOOK
- HI-FI - AUDIO - VIDEO - ACUSTICA
- ENERGIA - ENERGIE ALTERNATIVE E RINNOVABILI
- FREQUENZE SEGNALI RADIOAMATORIALI
- ANTENNE TV E SAT
- CB - RADIOAMATORE - ANTENNE
- TELECOMUNICAZIONI
- FISICA - CHIMICA - OTTICA
- MECCANICA
- ILLUMINOTECNICA
- ELETTROTECNICA
- IMPIANTISTICA
- SICUREZZA - NORME
- FAI DA TE
- ENERGIE FREQUENZE COSMICHE
- FORMULARI
- DIZIONARI
- INFORMATICA
- ELETTROMAGNETISMO
- RADIO D'EPOCA E SURPLUS
- STORIA RADIO - TV - ELETTRONICA
- INQUINAMENTO ELETTROMAGNETICO
- METEO
- DIVULGATIVI SCIENTIFICI
- ESPERIMENTI SCIENTIFICI
- REFRIGERAZIONE TECNOLOGIE DEL FREDDO
- SCIENZIATI, INVENZIONI, INVENTORI
- SCIENZA PER RAGAZZI E NON
- ECO - SALUTE
- RADIONICA - RADIOESTESIA
- SCIENZA E FANTASCIENZA
- AUTOCAD DISEGNO
- NOVITA' ULTIMI ARRIVI
- FUORI CATALOGO E SCONTATI

NOVITÁ! SANDIT LIBRI

IMPIANTI FOTOVOLTAICI
Come ricavare l'elettricità dal sole
G. Luoni
La costituzione dei pannelli fotovoltaici. Le caratteristiche dei pannelli fotovoltaici. Composizione di un sistema fotovoltaico. La batteria tampone. Il regolatore di carica. Convertitori statici. Chopper. Invertitore. Applicazioni fotovoltaiche. Alimentazione di utenze domestiche ed agricole.
Alimentazione di un impianto per il pompaggio dell'acqua. Tecniche di dimensionamento di un impianto. Installazione dei pannelli. Il costo dei chilowattora fotovoltaico. Sistemi di calcolo da sviluppare. Sistema di illuminazione per appartamento. Sistema di alimentazione energia elettrica per baita in montagna, per radiofaro, per colonnina S.O.S., per barca, per motore in C.A. Faro luminoso di segnalazione. Boa di segnalazione. Apparecchiatura
Pagine 48 **cod. 4293 - Euro 7,90**

SENSORI FOTOELETTRICI E DISPOSITIVI PER LA SICUREZZA
Concetti di base per una corretta scelta
G. Luoni
La luce. Trasmissione, assorbimento, riflessione. Il sensore fotoelettrico. Sensori fotoelettrici universali. Sbarramento a proiettore-ricevitore. Sbarramento a retroriflessione. Il riflettore prismatico. Sbarramento polarizzato. Sbarramento per trasparenti. Tasteggio diffuso. Tasteggio a soppressione di sfondo. Tasteggio a soppressione di primo piano e di sfondo. Sensori fotoelettrici applicativi. Sensori a forcella, di contrasto, di luminescenza, di colore, a fibra ottica, ad emissione laser. Dispositivi per la sicurezza. La direttiva macchine e la sicurezza. I sistemi di sicurezza con dispositivi optoelettronici. Valutazione del rischio e delle categorie di sicurezza. Sistemi di sicurezza di tipo 1, di tipo 4. Risoluzione e minimo oggetto rilevabile. Distanza di installazione. Funzione muting e override. Glossario termini tecnici. Fotoemettitori. Diodi led. Tasteggio diffuso. Valori limite-tensione inversa. Dissipazione massima. Massima corrente diretta. Massima temperatura di esercizio. Corrente diretta ad impulsi. Tensione diretta per una certa corrente. Intensità luminosa per una data corrente. Lunghezza d'onda della luce emessa. Angolo di apertura. Diodi laser. Fotorivelatori. Effetto elettrico. Fotodiodi. Fototransistor.
Pagine 48 **cod. 4294 - Euro 7,90**

SANDIT MARKET
Via Quarenghi, 42/c
24122 Bergamo
Tel. e Fax 035/321637
info@sandit.it
www.sanditmarket.it

SPESE DI SPEDIZIONE CON CONTRASSEGNO Euro 2,70

TELECONTROLLO GSM EL35.002:

UN "TELEASSISTENTE" AL VOSTRO SERVIZIO

Un prodotto che coniuga la tecnologia più avanzata con la semplicità, permettendovi di comandare apparecchiature remote con un semplice sms.

TELECONTROLLO GSM EL35.002: IL "TELEASSISTENTE"

Il Telecontrollo **GSM EL35.002** prodotto dalla **Elettroterm** è un mezzo sicuro ed affidabile per attivare a distanza impianti ed apparecchiature ad un costo molto competitivo.
Questo dispositivo si differenzia dalle apparecchiature complesse e sofisticate della gamma Elettroterm per via dello sforzo di semplificazione e razionalizzazione che è stato alla base della sua progettazione. Obiettivo dell'azienda è stato quello di poter proporre un dispositivo che potesse essere facilmente gestito ed utilizzato da chiunque sapesse inviare un messaggio sms con un telefono cellulare.
Lo scopo è stato raggiunto dotando l'apparecchio di tutta la componentistica e delle linee di ingresso e di uscita necessarie allo scopo, ma evitando di complicarlo con l'introduzione di funzioni destinate a modelli più complessi e sofisticati.
Il risultato è stato un eccezionale strumento che, pur essendo dotato della tecnologia più avanzata, riesce ad essere al contempo semplice, affidabile e competitivo.

A CHE COSA SERVE

Il Telecontrollo GSM EL35.002 vi permette di comandare a distanza, tramite il vostro cellulare, qualsiasi apparecchiatura o impianto collegato alla linea elettrica (o ad una batteria adeguata).
L'impiego tipico è nelle seconde case, specie se sprovviste di linea telefonica fissa (cosa che sta diventando sempre più comune a seguito della diffusione dei telefoni cellulari).
Collegando ad esempio il Telecontrollo EL35.002 al cronotermostato dell'impianto di riscaldamento, si potrà comandarne l'accensione o lo spegnimento mediante un semplice messaggio sms.
Il sistema consente evidentemente una grande libertà d'azione e una grande comodità di impiego: il comando sms può essere inviato da qualsiasi luogo e in qualsiasi momento.
Naturalmente, se d'inverno sarà l'impianto di riscaldamento ad essere attivato, d'estate si potrà invece scegliere ad esempio di telecomandare l'impianto di condizionamento, oppure lo scaldabagno elettrico in modo da trovare pronta l'acqua calda per una doccia tonificante all'arrivo.
I vantaggi sono evidenti. Oltre all'indubbio piacere di trovare la casa calda o fresca quando lo si vuole, si potranno anche ottenere sensibili risparmi sulla bolletta del gas o della luce, grazie alla possibilità di avviare gli impianti esattamente al momento voluto, senza inutili sprechi pur godendo ugualmente della massima comodità.
Naturalmente sono possibili moltissimi altri impieghi: basta che ci sia una apparecchiatura che si desideri far funzionare al momento voluto anche in nostra assenza.
Ad esempio si potrà telecomandare un qualsiasi elettrodomestico, quale un frigorifero che si voglia poter trovare con lo scomparto dei surgelati già in temperatura al momento del nostro arrivo, oppure una lavapiatti, o un

forno elettrico…
E non ci saranno dubbi o incertezze, perché il Telecontrollo EL35.002 provvederà a comunicare se l'impianto è acceso o spento.
Oltre agli impieghi casalinghi, anche altri settori possono beneficiare del telecontrollo: si potrà avviare la pompa dell'acqua per innaffiare il giardino o per irrigare i campi; nelle serre si potranno bagnare i fiori, o fornire dell'aria calda in inverno, o aprire delle finestre, o spostare delle tende o delle tapparelle…
Si potranno aprire le finestre di una mansarda per dare un po' d'aria in un giorno di canicola, o chiuderle in caso di pioggia; si potranno attivare impianti isolati come pozzi o centrali eoliche, fare scattare segnali, accendere impianti di illuminazione…
Per la sicurezza della casa potrete avviare la registrazione dei segnali di una telecamera, accendere le luci… Inoltre non avrete necessità di comunicare a terzi il codice del vostro impianto antifurto (dell'abitazione privata o dell'ufficio) perché potrete attivarlo da remoto dopo l'uscita del personale delle pulizie.
Nella versione EL35.002B, il dispositivo può essere azionato, anziché con un sms, con un semplice squillo telefonico a costo zero. Anche in questa versione il sistema prevede che, dopo aver eseguito il comando, venga emesso un sms di conferma indirizzato al telefono che ha inviato il comando stesso, però questa funzione può essere disattivata. Ciò risulta particolarmente utile nel caso della apertura automatica del cancello o del portone di casa che, essendo normalmente a vista, non necessita di messaggi di conferma per sapere se è chiuso o aperto. Poiché è possibile memorizzare quali numeri di telefono sono abilitati al comando, ciò rappresenta un importante fattore di SICUREZZA rispetto ai normali telecomandi per cancelli. Nel caso di condomini, dato che ormai praticamente tutti dispongono di un cellulare, potrà essere conveniente comandare con un telecontrollo EL35.002B il cancello condominiale, risparmiando sui telecomandi e garantendo meglio tutti gli inquilini contro gli ingressi indesiderati, grazie al funzionamento permesso solo ai numeri di telefono abilitati.

IL CUORE DEL PRODOTTO

Il "cuore" del Telecontrollo EL35.002 è un engine GSM Dual-Band di ultima generazione, in grado di colloquiare con qualsiasi telefono.
Al suo interno, come per tutti i telefoni cellulari, deve essere inserita una SIM Card, da utilizzare nel formato "plug-in" (vedi figura):
Tuttavia la SIM Card, prima del suo inserimento nel Telecontrollo, deve essere "programmata".
L'operazione è semplice: basta inserirla in un qualsiasi telefono cellulare e, seguendo le istruzioni del cellulare stesso, cancellare tutti gli eventuali numeri telefonici che vi sono memorizzati; dopo di che basterà inserire un codice che funzionerà da password a protezione del futuro impiego.

Figura 1: La SIM Card

A questo punto la SIM Card potrà essere tolta dal cellulare per essere inserita nel dispositivo EL35.002.

FUNZIONAMENTO

Il Telecontrollo EL35.002 è dotato di due led di avvertimento e di un pulsante per la commutazione manuale dell'uscita. Il primo led, di colore rosso, fornisce informazioni sulla presenza di campo sufficiente alla ricezione dei messaggi sms. Il secondo led, bicolore, quando è rosso indica che l'impianto comandato è spento, mentre quando è verde indica che l'impianto sta funzionando.
Durante l'installazione del dispositivo è importante verificare la presenza di campo. Il led rosso emette due diversi tipi di lampeggio in funzione del campo disponibile. Naturalmente prima dell'installazione è conveniente ricercare il punto di migliore ricezione nell'appartamento utilizzando un telefono cellulare. Se si rileva una carenza di segnale nel punto dove si vorrebbe fissare il Telecontrollo, è possibile utilizzare l'antennina opzionale a base magnetica, collegabile all'apparecchio mediante un cavetto di 2,5 metri, che consente di allontanarsi un po' dal punto di fissaggio.
La ricezione del campo viene migliorata se l'antenna viene fissata (magneticamente) ad un supporto metallico.
Una volta completate le operazioni preliminari, il dispositivo è pronto a funzionare.
I comandi sms da inviare (con un codice) sono semplicissimi:
- Accendi
- Spegni
- Verifica stato attuale

Come si può constatare, il dispositivo può essere descritto in sintesi con queste tre caratteristiche: semplice, affidabile, funzionale.

Per ulteriori informazioni:
ELETTROTERM srl
Via San Donato, 24 - 20052 Monza (MI)
Tel. 039-832465
info@elettroterm.it – www.elettroterm.it

Organizza

I° WORKSHOP SUL DEVICE NETWORKING

In collaborazione con: **SENA TECHNOLOGIES**

Sessione Mattutina (09.00-13.00)

EMBEDDED INTERNET

Il corso permetterà al partecipante di apprendere nozioni sulla configurazione ed integrazione dei dispositivi Sena Technologies (www.hellodevice.it) serie HD1x00, Serie Super SS110, SS400, SS800

Argomenti principali della sessione:

- problematiche di connessione di apparecchiature e dispositivi elettronici alle reti TCP/IP
- i micro web servers Sena Technologies ed i loro possibili campi di applicazione
- tecniche per il controllo remoto e via web
- domande e risposte

QUOTA DI PARTECIPAZIONE € 99,00 + IVA*
COMPRESO UN MICRO WEB SERVER
MOD. HD1100 DEL VALORE DI € 125,00 + IVA

Sessione Pomeridiana (14.00-18.00)

L'INTERFACCIAMENTO DEI DISPOSITIVI SERIALI ALLE RETI LAN

Il corso permetterà al partecipante di apprendere nozioni sulla configurazione ed integrazione dei dispositivi Sena Technologies (www.hellodevice.it) serie Lite, Pro, STS, VTS.

Argomenti principali della sessione:

- la conversione Seriale/Ethernet semplice ed economica
- bridge di dispositivi seriali mediante LAN
- interfacciamento di dispositivi con diversi livelli di integrazione (dal modulo all'apparato multiporta)
- il console management
- domande e risposte

QUOTA DI PARTECIPAZIONE € 99,00 + IVA*
COMPRESO UN SERIAL/ETHERNET CONVERTER
MOD. LS100 DEL VALORE DI € 125,00 + IVA

ISCRIVITI AD ENTRAMBE LE SESSIONI AL PREZZO IRRIPETIBILE DI € 189,00 +Iva
compresi 2 sistemi del valore complessivo di € 250,00 +IVA (1 mod. HD1100 + 1 mod. LS100)

elettroshop ACADEMY

SCHEDA DI PREREGISTRAZIONE

Da compilarsi in ogni sua parte ed inviare via fax al n. 02 66508225 o per e-mail ad academy@elettroshop.com
È possibile iscriversi online all'indirizzo www.elettroshop.com/academy

Evento di*: ❑ Roma ❑ Milano ❑ Padova ❑ Torino

Nome e Cognome ______ Azienda ______

Via ______ Cap ______ Città ______ Prov. ______

Tel. ______ Fax ______ E-mail ______

Sessione Workshop:

❑ Mattutina (€ 99,00 + Iva compreso un sistema HD1100)
❑ Pomeridiana (€ 99,00 + Iva compreso un sistema LS100)
❑ Entrambe (€ 189,00 + Iva compreso un sistema HD1100 e un LS100)

Firma

Autorizzo il trattamento dei dati personali ai sensi della Legge 675/96

*Seguiranno dettagli sulle date degli eventi (I Workshop si terranno nei mesi di Aprile/Maggio 2004) e sulle modalità del saldo della quota di partecipazione